



Huỳnh Phước Nam (Chủ biên)



Phương Lan (Hiệu đính)

UEF

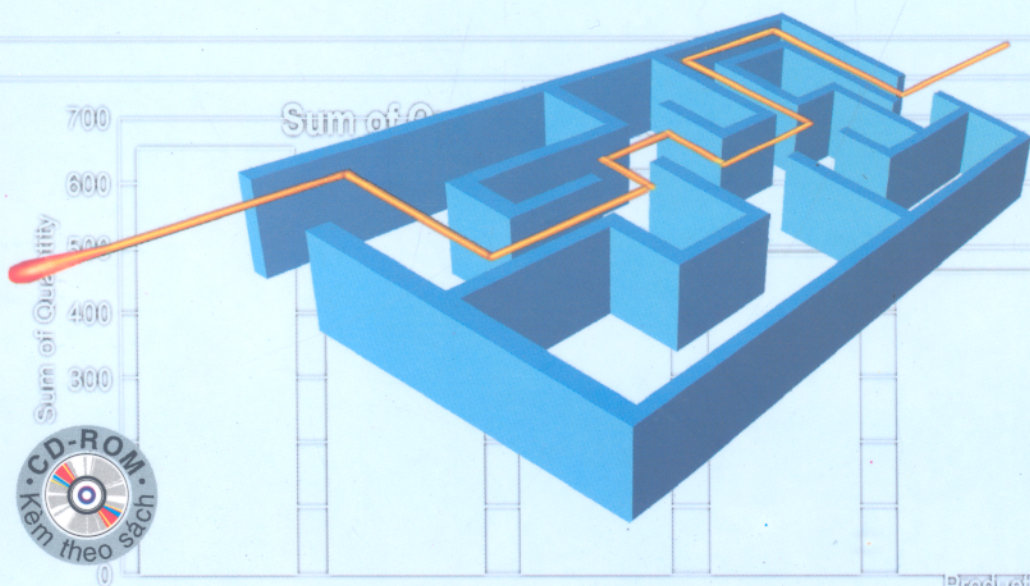
Sử dụng

Crystal Reports XI

Printed Date: 04/05/2005

modified: 04/05/2005

Report Description



NHÀ XUẤT BẢN PHƯƠNG ĐÔNG



Huỳnh Phước Nam (*Chủ biên*)
Phương Lan (*Hiệu đính*)

TIN HỌC VÀ ĐỜI SỐNG

Sử dụng

Crystal Reports XI

NHÀ XUẤT BẢN PHƯƠNG ĐÔNG

LỜI NÓI ĐẦU

Crystal Reports XI là một công cụ mạnh xử lý thông tin dữ liệu thô. Người dùng đầu cuối và các chuyên viên đều có thể cùng sử dụng Crystal Reports để trích xuất và định dạng thông tin. Với hơn 9 triệu bản Crystal Reports được bán ra và sử dụng trên toàn thế giới đã cho thấy đây là một công cụ được người dùng ưa chuộng nhất. Crystal Reports XI cung cấp chi tiết cách thiết kế báo biểu cho các ứng dụng phần mềm cũng như người mới bắt đầu học Crystal Reports. Các chuyên viên thiết kế đã từng làm việc với Crystal Reports lâu năm cũng sẽ thấy ở Crystal Reports phiên bản mới này nhiều tính năng mạnh mẽ.

Thực hiện từng bước theo sự hướng dẫn của giáo trình, bạn sẽ học cách tạo báo biểu từ nguồn dữ liệu và cách thêm các chức năng biểu diễn dữ liệu báo biểu một cách trực quan. Bạn cũng sẽ học được cách thực hiện tạo báo biểu từ đơn giản cho đến hoàn tất báo biểu thông qua định dạng và các kỹ thuật cấu hình trình bày báo biểu, lọc (Filter) dữ liệu, tính tổng gộp (Summary), sắp xếp hàng triệu dòng dữ liệu vào trong một khuôn dạng xúc tích, đầy ý nghĩa.

Bạn cũng sẽ học được cách tự tạo các công thức và câu lệnh SQL trong SubReport, Cross-Tab, biểu đồ (Chart), các biến động..

Với giáo trình này chúng tôi hy vọng rằng sẽ cung cấp cho bạn đọc các kỹ thuật cơ bản nhất để có thể thực hiện báo biểu như mong muốn.

Chúc các bạn thành công.

MK.PUB

THƯ NGỎ

Kính thưa quý Bạn đọc gần xa!

Trước hết, Ban xuất bản xin bày tỏ lòng biết ơn và niềm vinh hạnh được đồng hành cùng Bạn đọc nhiệt tình ủng hộ tủ sách MK.PUB.

Trong thời gian qua chúng tôi rất vui và cảm ơn các Bạn đã gửi e-mail đóng góp nhiều ý kiến quý báu cho tủ sách.

Mục tiêu và phương châm phục vụ của chúng tôi là:

- *Lao động khoa học nghiêm túc.*
- *Chất lượng và ngày càng chất lượng hơn.*
- *Tất cả vì Bạn đọc.*

Một lần nữa, Ban xuất bản MK.PUB xin kính mời quý Bạn đọc tiếp tục tham gia cùng chúng tôi để nâng cao chất lượng sách.
Cụ thể:

Trong quá trình sử dụng sách, nếu quý Bạn phát hiện thấy bất kỳ sai sót nào (*dù nhỏ*) xin đánh dấu, ghi chú nhận xét ý kiến của Bạn ra bên cạnh rồi gửi cuốn sách này cho chúng tôi theo địa chỉ:

Nhà sách Minh Khai

249 Nguyễn Thị Minh Khai, Q.1, Tp. Hồ Chí Minh.

E-mail: *mk.book@minhkhai.com.vn* hoặc *mk.pub@minhkhai.com.vn*

Chúng tôi xin hoàn lại cước phí bưu điện và gửi trả lại Bạn cuốn sách cùng tên. Ngoài ra chúng tôi còn gửi tặng Bạn một cuốn sách khác trong tủ sách MK.PUB. Bạn có thể chọn cuốn sách này theo danh mục thích hợp sẽ gửi tới Bạn.

Với mục đích ngày càng nâng cao chất lượng tủ sách MK.PUB, chúng tôi rất mong nhận được sự hợp tác nhiệt tình của quý Bạn đọc gần xa.

"MK.PUB cùng Bạn đọc đồng hành" để nâng cao chất lượng sách.

Một lần nữa chúng tôi xin chân thành cảm ơn.

MK.PUB

MỤC LỤC

LỜI NÓI ĐẦU	3
THƯ NGỎ	4
MỤC LỤC	5
Chương 1: GIỚI THIỆU VỀ CRYSTAL REPORTS	15
1. Tổng quan	15
2. Crystal Reports XI là gì?	16
3. Crystal Reports XI có gì mới?	17
3.1. Trình thiết kế báo biểu <i>Report Designer</i>	17
3.2. Các chức năng làm báo biểu <i>Report</i>	20
3.3. Xuất báo biểu (<i>Export</i>)	24
4. Crystal Reports XI dành cho nhà phát triển có gì mới?	25
4.1. Các nhà phát triển <i>.NET</i>	25
4.2. Các nhà phát triển <i>Java</i>	26
4.3. Các nhà phát triển ứng dụng khác	27
5. Phiên bản Crystal Reports XI	27
5.1. <i>Crystal Reports XI Standard Edition</i>	27
5.2. <i>Crystal Reports XI Professional</i>	28
5.3. <i>Crystal Reports XI Developer Edition</i>	28
6. Tóm tắt	28
Chương 2: TIẾP CẬN CRYSTAL REPORTS	29
1. Tổng quan	29
1.1. Cách truy xuất dữ liệu của <i>Crystal Reports</i>	29
1.2. Tạo một báo biểu <i>Report</i> trống	31
2. Tạo báo biểu <i>Report</i> với <i>Standard Report Wizard</i>	40
2.1. <i>Standard Report Wizard</i>	41

2.2. Blank Report Wizard.....	42
2.3. Cross-Tab Wizard.....	42
2.4. Mailing Label Wizard (tạo nhãn thư tự động).....	43
2.5. Online Analytical Processing (OLAP) Wizard.....	43
2.6. Chọn nguồn dữ liệu.....	44
2.7. Liên kết các bảng CSDL hoặc các File.....	45
2.8. Chọn trường dữ liệu.....	46
2.9. Nhóm và sắp xếp.....	47
2.10. Thêm phân tổng gộp.....	49
2.11. Sắp xếp nhóm.....	51
2.12. Thêm vào biểu đồ Chart.....	52
2.13. Tuyển chọn mẫu tin cho báo biểu.....	52
2.14. Chọn khuôn mẫu (Template).....	53
2.15. Kết hợp mọi thứ lại.....	54
3. Lưu báo biểu Report.....	56
3.1. Các tùy chọn lưu file.....	56
3.2. Định dạng file.....	57
3.3. Lưu dữ liệu trong cùng báo biểu Report.....	58
4. Làm việc với trình thiết kế báo biểu Report Designer.....	58
4.1. Môi trường Report Designer.....	58
4.2. Report Explorer.....	61
4.3. Tùy biến môi trường thiết kế.....	64
5. Tóm tắt.....	68

Chương 3: ĐỊNH DẠNG VÀ CẤU HÌNH BÁO BIỂU REPORT 69

1. Tổng quan.....	69
2. Tìm hiểu các đối tượng Field.....	70
2.1. Field CSDL (Database Field).....	70
2.2. Field tiêu đề (Heading Field).....	70

2.3. Field công thức (Formula Field).....	71
2.4. Field tham số biến (Parameter Field)	71
2.5. Field đặc biệt (Special Field).....	71
2.6. Các Field tổng chạy dồn (Running Total Field).....	71
2.7. Các Field biểu thức SQL (SQL Expression Field).....	72
2.8. Các Field tổng (Summary Field)	72
2.9. Field tên nhóm (Group Name Field).....	72
3. Chèn đối tượng Field vào báo biểu.....	72
4. Làm việc với các đối tượng Field	74
4.1. Định dạng các đối tượng Field	74
4.2. Điều chỉnh kích thước các Field	81
4.3. Di chuyển các đối tượng Field	82
4.4. Điều khiển các lớp (Layer)	84
4.5. Đối tượng Text	85
5. Cách hoạt động của các phân đoạn (Section)	88
5.1. Thêm một phân đoạn mới.....	89
5.2. Xóa một phân đoạn	90
5.3. Thay đổi kích thước phân đoạn.....	90
5.4. Tách phân đoạn	91
5.5. Trộn các phân đoạn báo biểu	91
5.6. Biến đổi thứ tự các phân đoạn	91
5.7. Định dạng các phân đoạn.....	92
5.8. Ẩn hoặc ngắt một phân đoạn báo biểu.....	93
5.9. Hiện thị các phân đoạn ẩn	94
5.10. In phân đoạn ở cuối trang	94
5.11. Tạo ngắt trang trước hoặc sau phân đoạn	94
5.12. Đặt lại số trang sau phân đoạn	95
5.13. Loại bỏ các phân đoạn trống	95

5.14. Tạo báo biểu nhiều cột.....	96
6. Thêm nhóm (Group) vào báo biểu.....	98
6.1. Tùy chọn định dạng nhóm.....	100
6.2. Tạo trang ngắt giữa các nhóm.....	104
6.3. Thay đổi nhóm.....	104
6.4. Nhóm theo chỉ định.....	108
6.5. Nhóm các Field dạng ngày hoặc ngày-giờ.....	109
7. Sắp xếp ở cấp mẫu tin.....	110
8. Drill-Down và báo cáo báo biểu Report.....	111
9. Tóm tắt.....	112
Chương 4: PHÂN TÍCH DỮ LIỆU BÁO BIỂU.....	113
1. Tổng quan.....	113
2. Tổng kết dữ liệu báo biểu.....	114
2.1. Dạng tổng kết (Summary) so với công thức (Formula).....	114
2.2. Thêm Field tổng kết trong báo biểu.....	116
3. Phân tích dữ liệu báo biểu.....	119
3.1. Sử dụng TopN/ BottomN phân tích trên nhóm.....	120
3.2. Sắp xếp theo tổng nhóm Subtotal hoặc Summary Field.....	121
3.3. Tổng chạy (Running Total).....	122
4. Sử dụng Record Selection để lọc mẫu tin trong báo biểu.....	126
4.1. Áp dụng lọc mẫu tin đơn giản dựa trên một Field.....	127
4.2. Áp dụng chọn lọc mẫu tin cho nhiều Field.....	129
4.3. Sử dụng phát biểu OR trong công thức chọn lọc mẫu tin.....	130
4.4. Áp dụng chọn lọc mẫu tin dựa vào những Field Date (Ngày tháng).....	131
4.5. Viết công thức chọn lọc mẫu tin.....	133
4.6. Chọn mẫu tin với những giá trị riêng biệt.....	134
4.7. Chọn mẫu tin dựa vào nhiều giá trị.....	135
5. Sử dụng tham số trong báo biểu.....	136

5.1. Làm việc với những Field tham số tĩnh.....	137
5.2. Chèn một Field tham số vào báo biểu.....	139
5.3. Thiết lập giá trị tham số mặc định.....	139
5.4. Sắp xếp các giá trị Field tham số mặc định.....	141
5.5. Nhập/Xuất các giá trị trong danh sách Field tham số.....	141
5.6. Tùy biến chuỗi nhắc tham số.....	143
5.7. Tạo Field tham số nhiều giá trị.....	143
5.8. Giới hạn các giá trị nhập của tham số.....	144
5.9. Sử dụng tham số tĩnh điều khiển các tính năng của báo biểu.....	145
5.10. Làm việc với tham số động.....	146
5.11. Sử dụng tham số động với giá trị lọc nối tầng.....	147
5.12. Tách tham số nối tầng thành những Field riêng biệt.....	149
5.13. Sử dụng đối tượng lệnh (Command) cho tham số động.....	150
5.14. Field tham số và Record Selection.....	152
5.15. Sử dụng một Field tham số lọc mẫu tin đơn giản.....	152
6. Tóm tắt.....	154
Chương 5: BIỂU THỨC, CÔNG THỨC VÀ HÀM	155
1. Tổng quan.....	155
2. Giới thiệu về công thức.....	156
2.1. Sử dụng Formula Editor.....	157
2.2. Cú pháp Basic và cú pháp Crystal.....	157
3. Cách hoạt động của công thức.....	158
3.1. Tạo một công thức mới.....	158
3.2. Chèn công thức vào trong báo biểu.....	159
3.3. Sửa công thức đã tồn tại.....	159
3.4. Đổi tên công thức.....	160
3.5. Tạo các công thức số học đơn giản.....	161
3.6. Thủ tục toán tử trong Crystal Reports.....	163

4. Sử dụng các hàm trong Crystal Reports	164
4.1. Các hàm tổng gộp.....	165
4.2. Hàm xử lý chuỗi.....	166
4.3. Các hàm chuyển đổi kiểu.....	167
4.4. Các hàm theo khoảng thời gian và Field ngày tháng.....	168
5. Tạo công thức logic	169
6. Thời điểm ước lượng công thức.....	171
6.1. BeforeReadingRecords.....	171
6.2. WhileReadingRecords.....	171
6.3. WhilePrintingRecords	172
6.4. EvaluateAfter	172
7. Bẫy lỗi công thức	172
8. Làm việc với hàm tùy biến.....	174
8.1. Tạo hàm tùy biến từ công thức có sẵn	174
8.2. Tự tạo hàm tùy biến.....	176
8.3. Sử dụng các hàm tùy biến trong công thức.....	178
9. Câu lệnh SQL và biểu thức.....	179
9.1. Cách hoạt động của câu lệnh SQL.....	179
9.2. Tạo câu lệnh SQL	180
9.3. Sửa câu lệnh SQL	181
9.4. Thêm tham số vào lệnh SQL.....	181
9.5. Cách hoạt động của biểu thức SQL.....	183
10. Tóm tắt.....	184
Chương 6: XÂY DỰNG BÁO BIỂU PHỨC HỢP	185
1. Tổng quan.....	185
2. Cách hoạt động của báo biểu con	185
2.1. Chèn Subreport không liên kết.....	188
2.2. Chèn một Subreport có liên kết.....	189

2.3. Kết hợp lại.....	190
2.4. Tạo một subreport theo yêu cầu.....	193
3. Định dạng báo biểu con Subreport.....	194
3.1. Thay đổi tên Subreport.....	194
3.2. Thay đổi đường viền.....	195
3.3. Thay đổi liên kết Subreport.....	196
4. Cách hoạt động của báo biểu ma trận (Cross-Tab).....	197
4.1. Tùy biến những tùy chọn nhóm (Group).....	201
4.2. Sắp xếp dữ liệu nhóm.....	202
4.3. Tùy biến tên Field nhóm.....	204
4.4. Làm việc với các Field tính gộp.....	205
4.5. Chèn các Field công thức.....	207
5. Định dạng Cross-Tab.....	208
5.1. Thay đổi Field, cột, dòng và kích thước.....	209
5.2. Sử dụng các kiểu định dạng có sẵn.....	209
5.3. Các kiểu định dạng tùy biến.....	210
6. Phân tích dữ liệu Cross-Tab.....	212
6.1. Sử dụng Highlighting Expert.....	212
6.2. Thay đổi hướng Cross-Tab.....	213
6.3. Công dụng và hạn chế của báo biểu Cross-Tab.....	214
7. Tóm tắt.....	214
Chương 7: BIỂU ĐỒ VÀ CÁC THÀNH PHẦN TRỰC QUAN.....	215
1. Tổng quan.....	215
2. Thêm biểu đồ vào báo biểu.....	217
2.1. Biểu đồ Advanced (Biểu đồ cấp cao).....	217
2.2. Biểu đồ Group (Biểu đồ nhóm).....	217
2.3. Biểu đồ Cross-Tab.....	218
2.4. Biểu đồ khung lưới OLAP.....	218

2.5. Thêm biểu đồ Advanced vào báo biểu.....	219
2.6. Thay đổi cách sắp xếp.....	222
2.7. Phân tích Top N (các giá trị đầu tiên).....	224
2.8. Thay đổi toán tử tổng gộp.....	225
2.9. Chèn biểu đồ nhóm.....	226
2.10. Chèn thêm biểu đồ Cross-Tab.....	229
2.11. Di chuyển biểu đồ.....	230
2.12. Thay đổi biểu đồ màu thành biểu đồ trắng và đen.....	231
2.13. Hiển thị chú giải.....	232
2.14. Điều khiển ô lưới và tỷ lệ đo.....	232
2.15. Thay đổi kiểu biểu đồ.....	233
2.16. Chỉnh kích thước Font và màu cho biểu đồ.....	233
3. Tóm tắt.....	234
Chương 8: TÍCH HỢP BÁO BIỂU	235
1. Tổng quan.....	235
2. Chọn phương pháp tích hợp	235
2.1. Report Viewer	235
2.2. Sử dụng Report Viewer với tham số.....	237
2.3. Tích hợp nâng cao.....	237
3. Lựa chọn nguồn dữ liệu.....	238
4. Chọn kiến trúc báo biểu	240
4.1. Ứng dụng Thick – Client.....	240
4.2. Ứng dụng Single –Tier	241
4.3. Ứng dụng Multitiered (đa tầng).....	241
5. Tóm tắt.....	242
Chương 9: TÍCH HỢP BÁO BIỂU VÀO ỨNG DỤNG WINDOWS. 243	
1. Tổng quan.....	243
2. Làm việc với mô hình Report View Object.....	245

2.1. Xem trước báo biểu.....	245
2.2. Sửa đổi diện mạo công cụ Viewer	246
2.3. In và kết xuất dữ liệu.....	248
2.4. Thay đổi điều kiện lọc mẫu tin của báo biểu.....	250
2.5. Làm việc với các Field tham số	254
2.6. Đăng nhập cơ sở dữ liệu	264
3. Làm việc với mô hình đối tượng ReportDocument	265
3.1. Chọn phương thức ràng buộc.....	265
3.2. In báo biểu.....	267
3.3. Xuất dữ liệu báo biểu.....	268
3.4. Làm việc với các tham số.....	271
3.5. Đăng nhập tự động vào cơ sở dữ liệu.....	276
4. Tóm tắt.....	278
Chương 10: TÍCH HỢP BÁO BIỂU VÀO ỨNG DỤNG WEB	279
1. Tổng quan.....	279
2. Tiếp cận ứng dụng Web	279
3. Làm việc với CrystalReportViewer	281
3.1. Xem báo biểu trên Web	281
3.2. Thay đổi cách thức hiển thị của Viewer.....	283
3.3. Chọn lựa chế độ in ấn.....	285
3.4. Hiệu chỉnh việc lọc các mẫu tin báo biểu.....	286
3.5. Làm việc với Field tham số.....	290
3.6. Đọc và thiết lập các giá trị rời rạc cho biến tham số.....	291
3.7. Cách thức đăng nhập CSDL.....	294
4. Tóm tắt.....	294
Chương 11: TÍCH HỢP BÁO BIỂU VÀO ỨNG DỤNG JAVA.....	295
1. Tổng quan.....	295
2. Tạo báo biểu sử dụng với ứng dụng JSP	296

2.1. Tìm hiểu JDBC	296
2.2. Làm việc với nguồn dữ liệu XML	300
3. Làm việc với các thành phần báo biểu Java	302
3.1. Cấu hình môi trường phát triển ứng dụng	302
3.2. Xem báo biểu trên trang Web	304
4. Làm việc với các Field tham số.....	306
5. Đăng nhập tự động vào CSDL.....	307
6. Tóm tắt.....	308
Chương 12: BUSINESSOBJECTS ENTERPRISE.....	309
1. Tổng quan.....	309
2. Tìm hiểu kiến trúc Server	310
3. Cài đặt BusinessObjects Enterprise XI	310
4. Xử lý báo biểu với đường dẫn URL	312
4.1. Xem báo biểu	312
4.2. Đăng nhập vào BusinessObjects Enterprise.....	313
4.3. Chọn lựa Viewer	314
4.4. Chuyển đổi định dạng báo biểu.....	316
4.5. Làm việc với các Field tham số	317
4.6. Lọc mẫu tin.....	318
4.7. Làm việc với CSDL	319
5. Tóm tắt.....	320

Chương 1:

GIỚI THIỆU VỀ CRYSTAL REPORTS

- ❖ *Tổng quan.*
- ❖ *Crystal Reports XI là gì?*
- ❖ *Crystal Reports XI có gì mới?*
- ❖ *Crystal Reports XI dành cho nhà phát triển ứng dụng có gì mới?*
- ❖ *Phiên bản Crystal Reports XI.*
- ❖ *Tóm tắt.*

1. TỔNG QUAN

“Báo biểu” hay còn gọi là “Report” đã trở thành một phần không thể thiếu trong quá trình phát triển ứng dụng. Báo biểu là kết quả kết xuất cuối cùng mà tất cả mọi người dùng chương trình của bạn mong đợi. Crystal Reports cũng được khai sinh gần như từ lúc có ứng dụng. Trong mỗi phiên bản của Microsoft® Visual Basic® và Visual Studio®, các nhà phát triển ứng dụng đã thể hiện sự hài lòng và không hài lòng của mình đối với sản phẩm Crystal Reports. Mặc dù các nhà phát triển ứng dụng trong môi trường của Microsoft rất thích các chức năng xây dựng báo biểu mà Crystal cung cấp nhưng đôi khi Crystal Reports cũng làm nản lòng các nhà phát triển vì sự tích hợp việc thiết kế báo biểu vào ứng dụng không được như ý. Hy vọng giáo trình Crystal Reports này sẽ hữu dụng cho bạn trong những trường hợp khó khăn khi xây dựng ứng dụng. Giáo trình này được soạn thảo chi tiết, rõ ràng và rất cần thiết cho các nhà phát triển ứng dụng, giúp họ có thể sử dụng Crystal Reports xây dựng chương trình kết xuất báo biểu với hiệu suất cao và ít tốn công sức nhất.

Giáo trình này được chia thành hai phần chính.

Nửa đầu giáo trình sẽ giới thiệu các tính năng thiết kế mẫu báo biểu, từ các chủ đề thiết kế đơn giản cho đến nâng cao. Đối với nhà phát triển ứng dụng, cho dù họ là người mới bắt đầu bước chân vào lĩnh vực sử dụng Crystal Reports hay đã có một số kinh nghiệm đối với Crystal Reports trước đó, giáo trình này cũng sẽ cung cấp đầy đủ nhất tất cả

những chức năng cần thiết của công việc xây dựng báo biểu với Crystal Reports, giúp người học có thể nắm được những khái niệm thiết kế báo biểu hiệu quả nhất thông qua rất nhiều chủ đề lý thú.

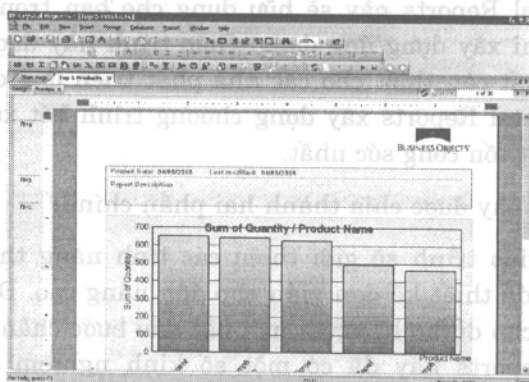
Phần còn lại của giáo trình sẽ hướng dẫn bạn các cách tích hợp báo biểu vào những công cụ phát triển ứng dụng khác nhau. Những đoạn mã ví dụ trong giáo trình này có thể là điểm bắt đầu hoặc là một khuôn mẫu khá tốt để bạn “bắt chước” làm theo nhằm đỡ tốn công sức trong quá trình phát triển ứng dụng có liên quan đến Report của riêng mình.

Các ứng dụng tích hợp báo biểu Crystal Reports có thể rất đa dạng, từ những chương ứng dụng Windows®, ứng dụng Web đến các ứng dụng Java™, trong phạm vi môi trường từ ASP.NET cho đến Java Server Pages.

Hầu hết các nhà phát triển ứng dụng đều đã có một ít kinh nghiệm thực tiễn với những công cụ phát triển và nền (platform) ứng dụng nào đó, đôi khi lại bị ép buộc sử dụng một công cụ hay nền tảng khác (như sự chuyển đổi giữa Java và .NET). Giáo trình này được biên soạn để cung cấp cho bạn cách thức tích hợp những mẫu báo biểu vào tất cả các nền như thế và một ngày nào đó trong cuộc đời phát triển ứng dụng bạn sẽ thực sự cần đến nó.

2. CRYSTAL REPORTS XI LÀ GÌ?

Crystal Reports XI là phiên bản nổi tiếng mới nhất dành cho các nhà phát triển ứng dụng chuyên thiết kế báo biểu (Report). Nó được xem là công cụ tạo báo biểu đầu tiên tích hợp với gói sản phẩm Visual Studio của Microsoft. Crystal Reports XI là phiên bản được phát hành vào đầu năm 2005 và nét đặc trưng là chức năng thiết kế báo biểu độc lập (hình 1.1). Bạn có thể dùng Crystal để thiết kế báo biểu từ nhiều nguồn dữ liệu khác nhau.



Hình 1.1: Giao diện Crystal Reports XI.

Một số hàm API trong Crystal Reports được tích hợp vào các gói ứng dụng như .NET, Java và ứng dụng COM. Thêm vào đó, các hàm API chuẩn phát triển trong phiên bản dành cho máy chủ Crystal Reports Server XI có thể dùng để tạo nên ứng dụng tùy biến, mạnh mẽ trên nhiều mức độ server và tầng xử lý khác nhau.

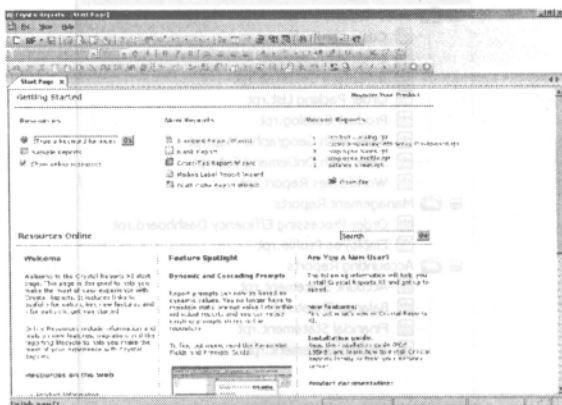
Ghi chú: Mã nền bên dưới kiến trúc Crystal Reports Server chính là BusinessObject™ Enterprise XI, một bộ khung để xây dựng và phân phối báo biểu với các tính năng chuyên nghiệp như bảo mật, lập lịch, đa cấu trúc v.v.... Để biết thêm thông tin về bản BusinessObject Enterprise XI bạn hãy vào trang web: www.businessobject.com/products/platform/enterprise.asp

3. CRYSTAL REPORTS XI CÓ GÌ MỚI?

Crystal Reports XI là phiên bản sản phẩm “đời” thứ 11, được cập nhật rất nhiều tính năng quan trọng và “có giá” hơn so với các phiên bản trước đó. Các chức năng mới được tổ chức kết hợp hài hòa và hợp lý hơn theo từng nhóm. Và hãy xem, có gì mới trong phần quan trọng nhất của Crystal, đó là sự xuất hiện của trình thiết kế báo biểu mang tên Report Designer.

3.1. Trình thiết kế báo biểu Report Designer

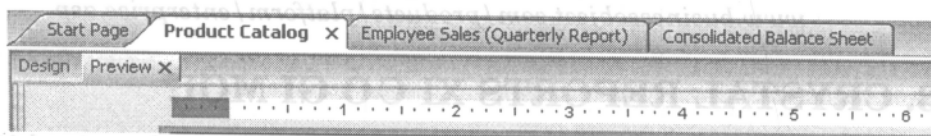
Nét đặc trưng của trình thiết kế Report Designer trong Crystal Reports XI là cập nhật lại giao diện đẹp hơn từ sự vay mượn một số thành phần của Microsoft Office. Điều đầu tiên mà bạn thấy khi mở ứng dụng Crystal Reports XI là một giao diện hoàn toàn mới so với các phiên bản trước đó (hình 1.2).



Hình 1.2: Trang đầu tiên của Crystal Reports XI.

Phần trên của trang giao diện là một hộp tìm kiếm nhanh, kể đến là một đường dẫn liên kết đến các ví dụ (Resource), phần tạo Report mới (New Reports) và mở các Report cũ (Recent Reports). Nửa phần còn lại là một trang HTML được kết nối trực tiếp đến trang Web BusinessObjects, trang này cũng chứa các đường dẫn liên kết trực tiếp đến nguồn tài nguyên của Crystal Reports. Bạn có thể tắt đi nguồn tài nguyên trực tuyến này bằng cách bỏ chọn hộp thoại **Show online resources** ở phần đầu trang.

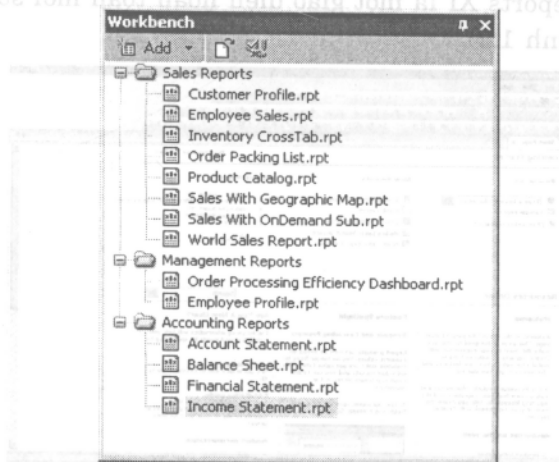
Nếu bạn mở nhiều Report cùng lúc, bạn sẽ thấy đồng thời tất cả các File Report. Mỗi file sẽ nằm trên một Tab riêng biệt mà các phiên bản Crystal Reports trước đây không có (hình 1.3).



Hình 1.3: Các Report mở cùng lúc trong trình thiết kế Report.

Tab thiết kế (Design) và Tab xem trước (Preview) vẫn còn, nhưng bây giờ được đặt ở phía dưới. Cách thiết kế này giúp dễ chuyển qua lại giữa các Report mà bạn đang làm việc, nhưng nó sẽ trở nên khó khăn đối với bạn nếu bạn đã quen sử dụng trình đơn (Menu) để chuyển qua lại giữa các Report và các tài liệu.

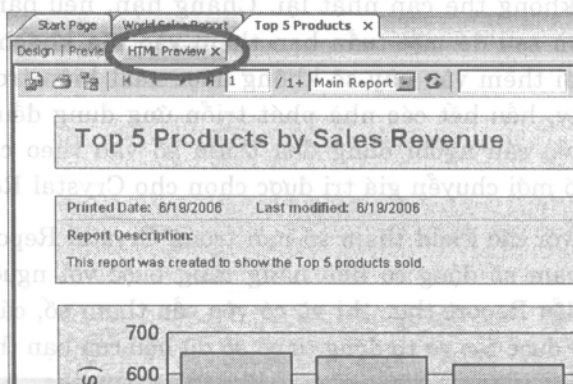
Một khái niệm mới khác trong Crystal Reports XI là WorkBench (Hình 1.4), chức năng này có thể dùng để thiết lập các báo biểu Report vào trong một dự án của ứng dụng (Project).



Hình 1.4: Giao diện WorkBench.

Khi bạn tạo một dự án, bạn có thể thêm vào nhiều báo biểu Report và xuất toàn bộ nội dung các dự án đến Crystal Reports Server hay BusinessObjects Enterprise XI để có thể xem trên Web.

Đối với Web, một trong các vấn đề mà các phiên bản Crystal Reports gặp phải khi thiết kế báo biểu cho ứng dụng Web là mẫu thiết kế trong trình Report Designer và kết quả khi xem trên Web sẽ không giống nhau. Đó là do báo biểu đã được chuyển sang dạng mã HTML, một số yếu tố đã không được biên dịch để thể hiện trên HTML (như góc hình tròn hay dòng văn bản). Để giải quyết các vấn đề trên, Crystal Reports XI đã tích hợp phần xem trước kết xuất HTML (hình 1.5) để bạn có thể thấy được sự khác biệt như thế nào giữa mẫu thiết kế và kết quả trên Web trước khi quyết định trình diện chúng.



Hình 1.5: Cửa sổ xem trước Report HTML.

Lợi ích của chức năng này là tiết kiệm được thời gian trong việc chuyển đổi qua lại giữa việc thiết kế Report và ứng dụng của bạn khi bạn không muốn gặp phiền phức trong việc chỉnh sửa kết quả sai lệch so với mẫu Report thiết kế. Bạn sẽ thấy ngay trang Report được thể hiện trên mã HTML như thế nào và sau đó có thể chuyển trở lại trang thiết kế Report hay trang Tab xem trước để thay đổi những thành phần giao diện không đúng.

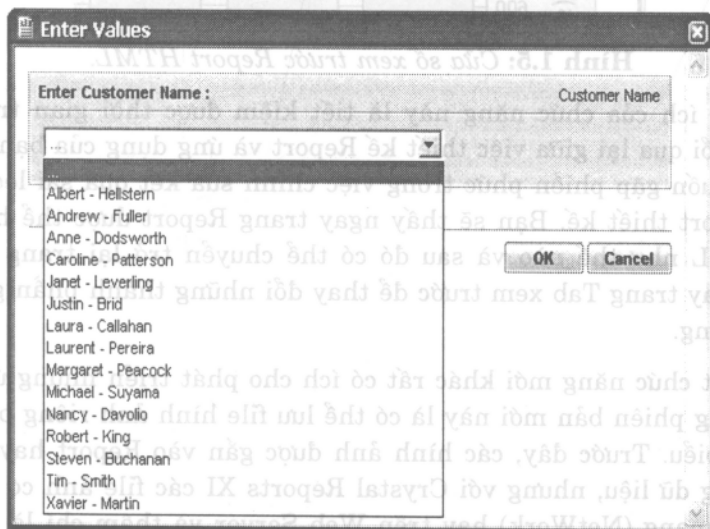
Một chức năng mới khác rất có ích cho phát triển những ứng dụng Web trong phiên bản mới này là có thể lưu file hình ảnh riêng bên ngoài file báo biểu. Trước đây, các hình ảnh được gắn vào Report hay lưu trực tiếp trong dữ liệu, nhưng với Crystal Reports XI các file ảnh có thể được lưu trên mạng (NetWork) hay trên Web Server và thậm chí là lưu ngay trong mẫu báo biểu Report của bạn.

Cuối cùng, nếu bạn trở nên phát chán và mệt mỏi trong việc giữ các bản sửa lỗi và cập nhật phiên bản mới nhất của Crystal Reports thì điều đó sẽ không còn nữa. Sản phẩm mới này đã tích hợp dịch vụ cập nhật tự sửa lỗi và cập nhật trực tuyến có thể tự cảnh báo đến bạn khi có các bản sửa lỗi (patch) từ nhà cung cấp.

3.2. Các chức năng làm báo biểu Report

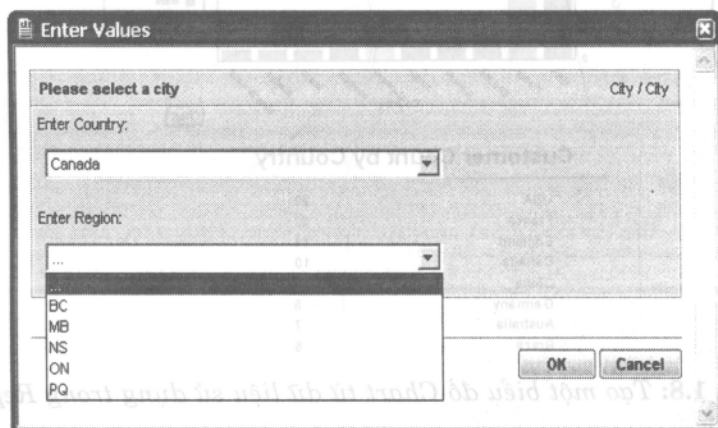
Chức năng xuất sắc nhất trong các chức năng làm báo biểu mà Crystal Reports cung cấp là khả năng tạo mới động các Field tham số. Trước đây, khi bạn tạo một Field hay biến trong Crystal Reports, danh sách giá trị chọn lựa mà bạn có thể thêm vào Report là tĩnh. Nói một cách khác, giá trị mà bạn thêm vào khi thiết kế báo biểu được lưu chung với Report và không thể cập nhật lại. Chẳng hạn, nếu bạn tạo tên biến **khách hàng** và sau đó một tuần bạn thêm vào tên khách hàng mới, vị khách hàng mới thêm vào này sẽ không được xuất hiện trong danh sách chọn. Trước đây, hầu hết các nhà phát triển ứng dụng đều phải tự thiết kế giao diện yêu cầu người dùng đưa tham số vào theo cách riêng của mình rồi sau đó mới chuyển giá trị được chọn cho Crystal Reports.

Bây giờ, với các Field tham số mới trong Crystal Reports XI, bạn có thể tạo biến tham số động có khả năng ràng buộc với nguồn dữ liệu của bạn. Khi báo biểu Report thực thi và có yêu cầu tham số, các giá trị trong danh sách chọn được tạo ra tự động từ cơ sở dữ liệu của bạn (hình 1.6).



Hình 1.6: Hoạt động của biến động.

Thêm vào đó, các tham số có thể được thiết lập nhiều tầng để giá trị biến tham số này có thể dùng để lọc giá trị biến tham số khác. Ví dụ, nếu bạn chọn tham số Enter Country là Canada thì chỉ có những tỉnh thành của Canada mới được xuất hiện trong danh sách tham số Enter Region (hình 1.7).

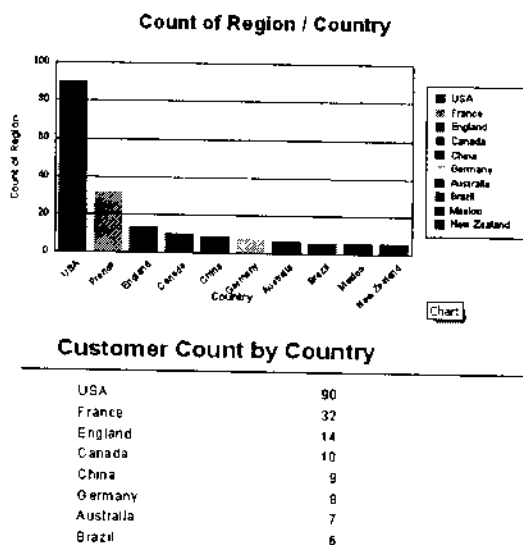


Hình 1.7: Ví dụ về tham số phụ thuộc.

Một tính năng mới khác đã có trong những phiên Crystal Reports trước đây được bổ sung mạnh hơn là tạo các báo biểu chéo (Cross-Tab) và biểu đồ (Chart) chỉ bằng một cú click chuột (Click-one). Trước đây, khi bạn muốn thêm Cross-Tab hay Chart vào báo biểu, một cửa sổ thông báo xuất hiện yêu cầu bạn chọn các trường hay Field thể hiện cho Cross-Tab hoặc tùy biến biểu đồ Chart. Sau đó, khi chọn xong bạn cần phải chọn tiếp vùng mà Cross-Tab hay Chart sẽ xuất hiện trong báo biểu. Tuy nhiên, nếu vùng mà bạn muốn thêm vào Cross-Tab hay Chart quá nhỏ so với nội dung thể hiện, bạn thường bị mất tất cả những gì mà công sức đã bỏ ra thiết kế trước đó và phải bắt đầu lại từ đầu.

Khái niệm Click-one có nghĩa là khi bạn thêm dạng báo biểu chéo Cross-Tab hay biểu đồ Chart, đối tượng này sẽ xuất hiện ngay trên đầu mũi tên con trỏ chuột, bạn có thể bấm chọn nó và đặt trên nền báo biểu của mình.

Trong trường hợp của Cross-Tab, một mẫu Cross-Tab rộng sẽ được hiển thị cho phép bạn có thể chỉnh sửa. Đối với loại biểu đồ Chart, Crystal Reports sẽ thử kiểu biểu đồ mà bạn ưng ý nhất. Nó có thể tạo ra biểu đồ cơ bản dựa trên thông tin mà bạn cung cấp, sau đó áp vào các thiết lập thuộc tính mặc định (hình 1.8).



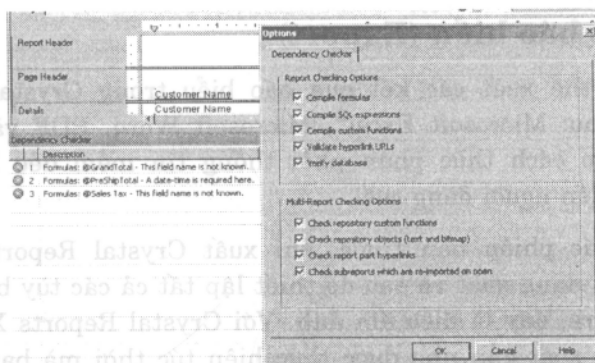
Hình 1.8: Tạo một biểu đồ Chart từ dữ liệu sử dụng trong Report.

Nếu Crystal Reports không thể đoán ra kiểu biểu đồ mà bạn muốn tạo, trình trợ giúp Chart Expert sẽ hiện ra, cho phép bạn chọn các trường mục Field tạo nên Chart và các thành phần khác. Để dễ dàng định dạng biểu đồ Chart, Crystal Reports cung cấp thêm một thanh menu xuất hiện khi bạn nhấp vào một đối tượng Chart trên báo biểu.

Một tính năng khác được quan tâm khá nhiều trong phiên bản này là nhóm phân cấp (Hierarchical group). Nhóm phân cấp làm việc dựa trên ý tưởng về cấu trúc dữ liệu cha con. Ví dụ, bảng nhân viên (Employees) có thể có mã ID liên quan đến người quản lý Manager tương ứng hoặc mỗi bảng sổ cái kế toán có chứa các tài khoản liên quan đến nhiều sổ phụ ở những bảng khác.

Trước đây, Crystal Reports đã có chức năng phân cấp nhóm nhưng nó không được linh động lắm. Hầu hết các nhà phát triển ứng dụng chọn cách sử dụng kết hợp nhóm bình thường (Normal group) và Field công thức (Formula Field) để giả lập chức năng phân cấp nhóm và họ có thể tự điều khiển cách dàn trang của báo biểu như thụt đầu dòng hay định dạng theo nhóm.

Crystal Reports XI đưa ra một tập công cụ mới cho phép làm việc với nhóm phân cấp, chúng bao gồm định dạng theo điều kiện kiểm soát kết xuất dàn trang của báo biểu, xác định vị trí đối tượng và canh lề cho phép tạo ra các Report phức tạp trong thời gian ngắn nhất.



Hình 1.9: Dependency Checker.

Chức năng cuối cùng này có lẽ là một vị anh hùng không được công nhận của Crystal Reports XI – nó mang tên Dependency Checker. Trong mỗi phiên bản, bẫy lỗi (debug) quá trình thực thi của báo biểu là một công việc khủng khiếp. Khi xây dựng công thức, bạn phải kiểm tra các Field công thức một cách cẩn thận để đảm bảo rằng chúng hoạt động tốt. Đôi khi, nếu muốn tìm đến những Field công thức bị thiếu hoặc các Field bị thay đổi trong lúc chạy báo biểu Report bạn sẽ phải nhờ đến các câu thông báo lỗi và theo dõi lỗi; hoặc các phần tử của công thức Field không chính xác và sửa lại cho chính xác để chạy Report.

Với bước đầu làm quen Dependency Checker (hình 1.9), bạn có thể kiểm tra sự tin cậy của Report để đảm bảo rằng báo biểu Report của bạn có thể chạy ngay ở lần đầu tiên.

Việc kiểm tra có thể được thực hiện sau đó để kiểm tra báo biểu Report của bạn như sau :

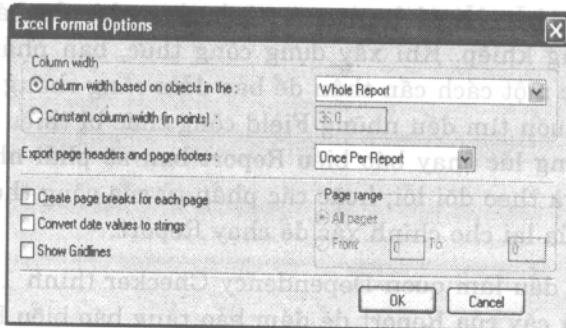
- Biên dịch các công thức.
- Biên dịch các biểu thức SQL.
- Biên dịch các hàm tùy biến.
- Xác nhận các đường dẫn URL.
- Kiểm tra cơ sở dữ liệu.

Thêm vào đó, Dependency Checker cũng có thể được dùng để kiểm tra các báo biểu Report con được du nhập (import) lại vào báo biểu chính. Chức năng mới này của Crystal Reports XI tiết kiệm thời gian của bạn khi làm việc với những báo biểu phức tạp và có thể đảm bảo rằng các Report được tích hợp vào ứng dụng sẽ chạy đúng đắn ngay lần đầu.

3.3. Xuất báo biểu (Export)

Bạn có thể xuất các kết quả báo biểu trong Crystal Reports ra nhiều dạng như: Microsoft Excel®, Microsoft Word, PDF và HTML. Nó cung cấp thêm cách thức phân phối thông tin mềm dẻo khác lưu trữ trong Report đến người dùng cuối.

Trong các phiên bản trước, khi xuất Crystal Reports, bạn phải chọn một định dạng xuất và sau đó thiết lập tất cả các tùy biến cho định dạng cần đưa ra, đây là điều ám ảnh. Với Crystal Reports XI, việc thiết lập định dạng xuất gần như được thực hiện tức thời mà bạn không cần phải thiết lập thêm tham số gì nữa. Ví dụ, bạn chỉ cần thiết lập tùy biến mặc định cho việc xuất kết quả ra Excel bằng cách sử dụng hộp thoại như hình 1.10. Lần tiếp theo khi bạn muốn xuất ra định dạng này, tùy biến xuất sẽ tự động được thiết lập cho bạn.

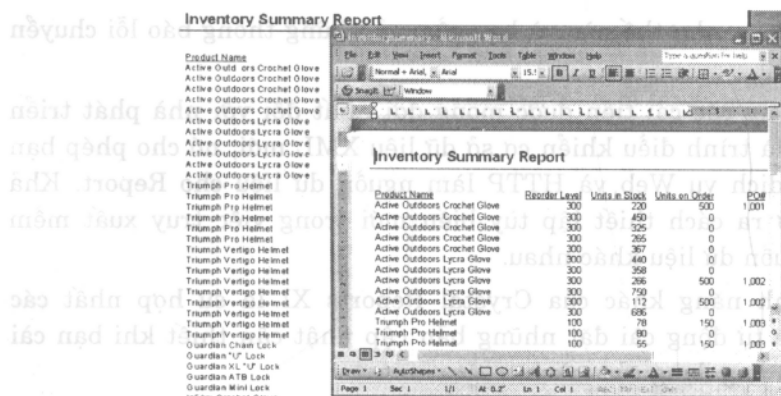


Hình 1.10: Thiết lập tùy biến xuất ra Excel.

Một tính năng nâng cao khác của việc kết xuất là có thể yêu cầu xuất báo biểu của bạn ra dạng có thể chỉnh sửa được .RTF. Khi bạn xuất một Report từ các phiên bản của Crystal Reports trước đây sang định dạng .RTF, thông tin trong Report thường hiển thị ở dạng ô TextBox trong file .RTF, người dùng thường gặp phải khó khăn trong việc sửa chữa thông tin hoặc sao chép và dán chúng vào tài liệu khác.

Với trình định dạng biên tập xuất .RTF mới, bạn có thể tạo ngay .RTF file và có thể sửa ngay nội dung báo biểu trong Word, WordPad hay một số chương trình soạn thảo văn bản khác. Hơn nữa, nội dung trong Report cũng dễ dàng được sao chép và dán vào những ứng dụng khác (hình 1.11).

Người dùng có thể biên tập nội dung báo biểu, thêm các chú thích và văn bản trong bản xuất Report, sau đó có thể dùng nó để tạo những báo biểu quản lý phức tạp khác.



Hình 1.11: Xuất Report ra định dạng .RTF mới.

4. CRYSTAL REPORTS XI DÀNH CHO NHÀ PHÁT TRIỂN CÓ GÌ MỚI?

Một tính năng mới được thêm vào cho các nhà phát triển trong bản Crystal Reports XI này đó là việc nâng cao khả năng tích hợp Report vào môi trường phát triển ứng dụng diễn ra dễ dàng và nhanh chóng hơn.

Trong phần tiếp theo, chúng ta sẽ xem xét những tính năng tích hợp được chia nhóm theo môi trường phát triển. Bắt đầu từ chức năng mới dành cho những nhà phát triển ứng dụng trong môi trường Visual Studio .NET.

4.1. Các nhà phát triển .NET

Hầu hết các nhà phát triển ứng dụng .NET đã quá quen thuộc với Crystal Reports từ phiên bản Visual Studio .NET 2002 và Visual Studio .NET 2003. Bắt đầu từ phiên bản Crystal Reports XI, C# Bulder® và Delphi 2005 đã được Crystal Reports xây dựng tích hợp hoàn hảo khiến nó là một trong những lý do đáng giá để nâng cấp các bộ công cụ phát triển ứng dụng này.

Cả Crystal Reports WinForm và WebForm cũng đã được cập nhật để hỗ trợ cho tất cả những chức năng có trong Crystal Reports XI như khả năng tạo biến tham số động, thay đổi giá trị tham số, biến, cập nhật định dạng xuất. Giao diện trông hấp dẫn hơn và bạn có thể tắt đi tính năng hướng dẫn khó chịu luôn xuất hiện trong các phiên bản trước. Khả năng tìm kiếm trên trình Viewer của WinForm đã được thay thế bằng một hộp thoại Find và thêm bộ xử lý sự kiện lỗi giúp bạn có thể xử lý các lỗi xảy ra trong quá trình thực thi báo biểu Report. Phần cải tiến khác là các câu thông báo lỗi được mô tả rõ ràng hơn cho biết cụ thể các lỗi nào

đã xảy ra, xảy ra như thế nào và bao gồm cả những thông báo lỗi chuyển từ cơ sở dữ liệu.

Một trong các cải tiến được mong đợi nhất đối với nhà phát triển ứng dụng đó là trình điều khiển cơ sở dữ liệu XML mới, nó cho phép bạn sử dụng các dịch vụ Web và HTTP làm nguồn dữ liệu cho Report. Khả năng này mở ra cách thiết lập tùy biến mới trong việc truy xuất mềm dẻo nhiều nguồn dữ liệu khác nhau.

Một tính năng khác của Crystal Reports XI là sự hợp nhất các module, nó sẽ tự động cài đặt những bản cập nhật cần thiết khi bạn cài đặt sản phẩm ứng dụng của mình.

4.2. Các nhà phát triển Java

Crystal Reports theo truyền thống thường dành cho những nhà phát triển ứng dụng Microsoft, nhưng trong vài năm trở lại đây, trường phái Java đã làm nên công nghệ đột phá mới. Phiên bản Crystal Reports OEM đã có sự thay đổi hỗ trợ cho môi trường Java IDE, bao gồm các trình phát triển ứng dụng như Jbuilder®, BEA® WebLogic Workshop™ và IBM® Rational®.

Đối với các nhà phát triển ứng dụng Java, chức năng Java Reporting Component (JRC) đã được Crystal Reports XI cập nhật, nhưng một điều đáng tiếc là không có tính năng tạo động tham số trong phiên bản này. Nhiều tính năng cải tiến cho trình phát triển ứng dụng Java đã được thêm vào, bao gồm điều khiển dữ liệu XML để tạo các báo biểu từ những ứng dụng Web Service, điều này giúp cho nhà phát triển ứng dụng làm việc theo phong cách hướng đến dịch vụ tốt hơn.

Các tập hàm Java API cũng đã được cải tiến cho phép nhà phát triển ứng dụng thiết lập nguồn dữ liệu ngay tức thời, thiết lập vị trí dữ liệu và kiểm tính đúng đắn của cơ sở dữ liệu.

Một cải tiến quan trọng khác là việc giới thiệu thư viện hàm người dùng Java (Java User Function Libraries) viết tắt là (UFL). UFL có thể làm nổi trội các chức năng đã được xây dựng trong Crystal Reports. Các thư viện chức năng này giống như thư viện chức năng của trình phát triển ứng dụng Windows đã được tạo sẵn trong ứng dụng Windows và ứng dụng Web thông qua việc sử dụng các file .DLL. UFL giúp bạn dễ dàng tạo nên các chức năng mà bạn có thể dùng lại trong các Report khác.

Crystal cũng hỗ trợ một thành phần Viewer dành cho JavaServer Faces. Nó cho phép tích hợp Crystal Reports vào ứng dụng đã tồn tại bằng cách thiết lập thẻ JSF tùy biến. Thẻ này là cú pháp tắt và nhanh

chóng nhất giúp các nhà phát triển ứng dụng đưa báo biểu của Crystal Reports vào ứng dụng Java mà không cần phải học các hàm API khác.

4.3. Các nhà phát triển ứng dụng khác

Crystal Reports XI cũng bao gồm một phiên bản cập nhật cho thành phần thiết kế Report (**Report Design Component** viết tắt là RDC). RDC là sự kế thừa các tập hàm API của Crystal Reports cung cấp ở dạng COM và dành cho những nhà phát triển trong ngôn ngữ Visual Basic cũ và không khuyến khích sử dụng trong những dự án mới. Công nghệ này đã được Crystal Reports sử dụng một vài năm nay trong Visual Basic 6, Delphi và ASP nhưng nó thật sự gần như đã chấm dứt vai trò của mình.

Chức năng mới RDC trong phiên bản 11 của Crystal Reports tập trung chủ yếu để tương thích. Nó giữ cho các ứng dụng được tạo trong các phiên bản trước của RDC vẫn hoạt động. Một vài chức năng trong trình phát triển khác không còn chạy được trong phiên bản này, đó là ứng dụng Server API. Server API là một phần của Crystal Reports Server và là chức năng mới được giới thiệu trong Crystal Reports XI.

5. PHIÊN BẢN CRYSTAL REPORTS XI

Bạn đã lướt qua một số tính năng mới của Crystal Reports XI, hãy sẵn sàng chạy ra cửa hàng và mua một bản. Nhưng trước khi dự định mua bạn hãy xem qua các phiên bản của Crystal Reports, nó được chia thành 4 phiên bản khác nhau như sau:

- Crystal Reports XI Standard Edition.
- Crystal Reports XI Professional.
- Crystal Reports XI Developer Edition.
- Crystal Reports XI Server.

Bộ “cơ khí” (engine) điều khiển Crystal Reports và môi trường thiết kế Report Designer cơ bản là giống nhau ở tất cả các phiên bản nhưng mỗi một phiên bản có một số chức năng của riêng nó. Các phiên bản theo sau luôn tốt hơn các phiên bản trước đó.

5.1. Crystal Reports XI Standard Edition

Crystal Reports XI Standard Edition bao gồm bản quyền sử dụng trình Report Designer và có thể sử dụng để tạo các Report từ các kiểu cơ sở dữ liệu của PC bao gồm Microsoft Access, dBase™ và v.v... Với phiên bản này bạn có thể tích hợp các báo biểu Report vào ứng dụng nhưng nó không có những thành phần công cụ (component) dành cho bản Developer và Server.

5.2. Crystal Reports XI Professional

Crystal Reports XI Professional bao gồm bản quyền sử dụng trình Report Designer và có thể sử dụng để tạo các Report từ các nguồn dữ liệu khác nhau thông qua ODBC (Open Database Connectivity), điều khiển cơ sở dữ liệu truyền thống v.v... Crystal Reports XI Professional Edition cũng là sự mở đầu cho việc giới thiệu Crystal Reports Server với bản quyền cho 5 người sử dụng. Bản quyền này cho phép bạn xuất Report lên Crystal Reports Server và 5 người sử dụng có thể kết nối lên Crystal Reports Server để xem và tương tác với các Report đó. Phiên bản này cũng không có công cụ Developer hoặc API tích hợp sẵn.

5.3. Crystal Reports XI Developer Edition

Crystal Reports XI Developer Edition bao gồm bản quyền sử dụng trình Report Designer và bản chuyên nghiệp (Professional). Và giống phiên bản Professional, bạn cũng có thể tạo Report từ các nguồn dữ liệu khác và nguồn dữ liệu lúc thực thi (runtime) gồm: ADO, RDO, CDO... Phiên bản này cung cấp tất cả các thành phần và công cụ mà bạn cần để tích hợp Crystal Reports vào ứng dụng.

Ấn bản này cung cấp thư viện thực thi runtime miễn phí cho người dùng sử dụng ứng dụng Windows, với phiên bản runtime này họ có thể phân phối trong phạm vi cho phép của cơ quan mình bao gồm các ứng dụng chạy nhiều bộ xử lý hay nhiều server. Nếu bạn là nhà phát triển ứng dụng sử dụng Crystal Reports để chia sẻ hay bán cho cơ quan hoặc tổ chức khác, bạn phải mua bản quyền phiên bản Crystal Reports XI Developer Edition cho mỗi sản phẩm phân phối.

Developer Edition cũng gồm một bản Crystal Reports Server với bản quyền 5 người dùng. Đây là phiên bản cho phép xuất Report lên Crystal Reports Server và 5 người dùng có thể kết nối để xem cũng như thao tác trên Report của họ. Crystal Reports Server cũng cung cấp sức mạnh cho những dịch vụ tạo ứng dụng đa báo biểu với cơ chế bảo mật và lập lịch được xây dựng sẵn.

6. TÓM TẮT

Chúng ta đã xem qua tóm tắt các chức năng mới của Crystal Reports XI và tập hàm API mà bạn có thể tích hợp vào ứng dụng của mình. Bây giờ, bạn và chúng tôi hãy cùng nhau xắn tay vào khám phá từng phần của Crystal Reports XI. Chương kế tiếp sẽ thú vị hơn khi chúng ta bắt đầu thực hiện công việc kết nối dữ liệu và tạo báo biểu Report đầu tiên của mình.

Chương 2:

TIẾP CẬN CRYSTAL REPORTS

- ❖ *Tổng quan.*
- ❖ *Cách truy xuất dữ liệu của Crystal Reports.*
- ❖ *Tạo báo biểu Report thông qua Standard Report Wizard.*
- ❖ *Lưu báo biểu Report.*
- ❖ *Vận hành của trình thiết kế báo biểu Report.*
- ❖ *Môi trường thiết kế báo biểu Report.*
- ❖ *Tùy biến môi trường thiết kế.*
- ❖ *Tóm tắt.*

1. TỔNG QUAN

Bất chấp các loại dữ liệu mà bạn muốn tạo báo biểu Report, Crystal Reports đều có thể khởi tạo kết nối đến nhiều nguồn dữ liệu khác nhau. Đó là một trong các sức mạnh của Crystal Reports, nó có thể tạo báo biểu từ những nguồn cơ sở dữ liệu như Desktop và file cấu trúc dữ liệu cho đến kho dữ liệu, cơ sở dữ liệu quan hệ, cơ sở dữ liệu OLAP và hệ thống ERP như SAP®, Oracle® JD Edwards® ... Trong phần đầu chương này, chúng ta sẽ học cách truy xuất dữ liệu của Crystal Reports, những phương pháp liên quan đến truy xuất dữ liệu và tạo báo biểu Report từ các nguồn dữ liệu này.

Phần cuối của chương này chúng ta sẽ học chi tiết về cách thiết kế báo biểu Report. Bạn sẽ học cách tạo báo biểu Report nhanh nhất bằng cách sử dụng trình Wizard trong Crystal Reports. Và sau cùng bạn sẽ học những tính năng của môi trường thiết kế Report như xem trước Preview và lưu lại báo biểu Report sau khi thiết kế.

1.1. Cách truy xuất dữ liệu của Crystal Reports

Crystal Reports có thể truy xuất nguồn dữ liệu theo 2 cách: thông qua một lớp chuyển đổi như ODBC hay thông qua trình điều khiển kết nối trực tiếp (native) từ Crystal Reports đến dữ liệu của bạn dựa vào những giao thức đặc thù của nguồn dữ liệu bạn muốn kết nối.

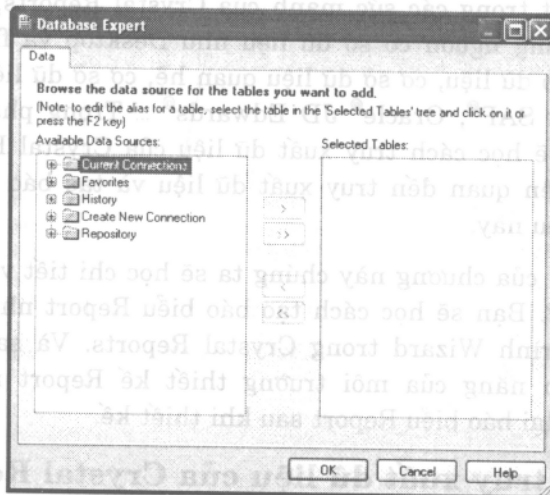
Qua nhiều năm, Crystal Reports đã kết hợp với nhiều loại cơ sở dữ liệu, ứng dụng và cung cấp khá nhiều trình điều khiển truy xuất trực tiếp (native) đến các kiểu cơ sở dữ liệu dạng file, các cơ sở dữ liệu quan hệ và hệ thống ERP trong đó có cả những trình điều khiển truy xuất Microsoft SQL Server, Oracle và nhiều hệ cơ sở dữ liệu nổi tiếng khác.

Ghi chú: Thông qua giao thức của những trình điều khiển native, bạn có thể truy xuất nguồn dữ liệu không chuẩn giống như file nhật ký, Microsoft Outlook® và thậm chí là chính hệ thống file.

Cách truy xuất thứ hai là sử dụng lớp chuyển đổi mà lớp chuyển đổi thường được sử dụng thông dụng nhất là ODBC. ODBC là giao thức phổ biến cho việc truy vấn quan hệ cơ sở dữ liệu. Bất chấp nơi đặt cơ sở dữ liệu là ở đâu, ODBC cũng cung cấp phương thức điều khiển và kết nối cơ sở dữ liệu vững chắc mà các nhà phát triển có thể tin tưởng sử dụng.

Chọn sử dụng cách kết nối trực tiếp native hay ODBC là tùy vào ứng dụng của bạn. Các lập trình viên vẫn thích dùng cơ chế native hơn vì nó loại bỏ lớp chuyển đổi trung gian mà ODBC yêu cầu.

Ghi chú: Trong bản thương mại của ODBC, Crystal Reports đã bao gồm phân điều khiển miễn phí đảm trách cho các cơ sở dữ liệu thông dụng.



Hình 2.1: Crystal Reports database Expert.

Tất cả các nguồn dữ liệu được đề cập ở trên đều có thể truy xuất thông qua Database Expert (hình 2.1). Database Expert sẽ hiển thị khi bạn

chọn nguồn dữ liệu để tạo báo biểu Report mới hoặc khi bạn thêm một cơ sở dữ liệu mới hay băng dữ liệu vào báo biểu Report hiện hành. Phần tiếp theo sau sẽ mô tả các kiểu dữ liệu thông dụng nhất cũng như một số chỉ dẫn cơ bản để bạn sử dụng chúng trong báo biểu Report của mình.

Ghi chú: Các báo biểu Report được sử dụng trong chương này đều nằm trong thư mục *Projects* trên đĩa CD-ROM kèm theo sách. Bạn có thể tự tạo lại các báo biểu Report trong chương hoặc chép các ví dụ trong đĩa CD-ROM vào ổ cứng của bạn để biên tập lại.

1.2. Tạo một báo biểu Report trống

Trong phần này chúng ta sẽ khảo sát các nguồn dữ liệu có thể sử dụng trong báo biểu Report của bạn và các kiểu định dạng dữ liệu đặc thù có thể truy xuất. Đối với mỗi nguồn dữ liệu, chúng tôi sẽ hướng dẫn bạn từng bước cách kết nối dữ liệu và tạo ra một báo biểu Report trống từ nó.

1.2.1. Cơ sở dữ liệu dạng File

Crystal Reports có thể sử dụng phương pháp kết nối trực tiếp native cho báo biểu Report từ các kiểu dữ liệu cấu trúc cơ bản sau đây:

- Microsoft Excel (.xls)
- XML (.xml)
- Xbase (.dbf, .ndx, .mdx, .bde)
- Paradox® (.db)
- Pervasive® PSQL V8 (.ddf)
- Microsoft Access (.mdb)

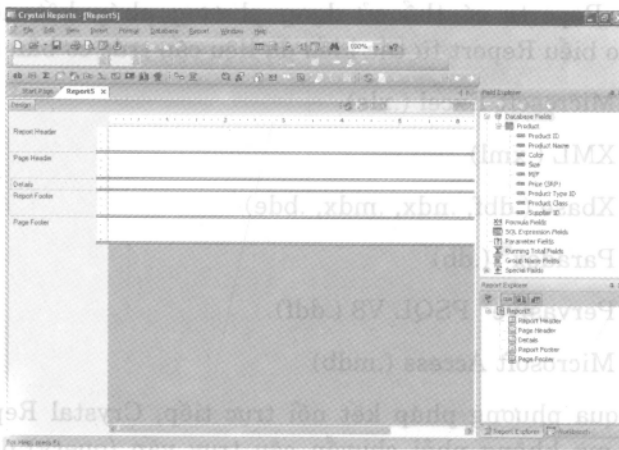
Thông qua phương pháp kết nối trực tiếp, Crystal Reports có thể trích dữ liệu mà không phải chuyển câu truy vấn (query) đến cơ sở dữ liệu Server. Để tạo một báo biểu Report từ nguồn dữ liệu kiểu cấu trúc này, bạn có thể bắt đầu từ việc tạo một báo biểu Report trống hoặc sử dụng một trong các kiểu báo biểu do trình trợ giúp Report Wizard sẽ được chúng ta tìm hiểu trong chương này.

Để tạo một báo biểu Report từ một trong các nguồn dữ liệu được Crystal Reports hỗ trợ trên đây phải qua các bước sau:

1. Mở Crystal Reports và chọn File > New > Blank Report. Cửa sổ Database Expert sẽ hiển thị.

2. Nhấp 2 lần vào mục Create New Connection, danh sách các nguồn dữ liệu được hỗ trợ sẽ hiển thị.
3. Nhấp 2 lần vào mục Database Files. Sau đó sử dụng hộp thoại File > Open để chỉ đến vị trí file cơ sở dữ liệu của bạn và nhấp chọn nó, sau đó nhấn OK. Trong ví dụ này, bạn hãy chọn cơ sở dữ liệu GALAXY.MDB trên CD-ROM.
4. Danh sách các trường của bảng sẽ xuất hiện ngay bên dưới nguồn dữ liệu của bạn. Bạn đánh dấu chọn bảng dữ liệu dự định sử dụng trong báo biểu Report bằng cách nhấp chọn và sau đó nhấn vào nút mũi tên chỉ về bên phải để di chuyển chúng qua danh sách bên kia. Đối với báo biểu Report mẫu mà chúng ta sắp tạo bạn hãy chọn bảng INVOICE.
5. Khi đã hoàn thành chọn bảng dữ liệu, nhấn OK để quay về cửa sổ thiết kế báo biểu Report.

Một báo biểu Report mới sẽ được tạo và hiển thị trong cửa sổ thiết kế báo biểu Report như hình 2.2.



Hình 2.2: Báo biểu Report trống được tạo ra từ một file cơ sở dữ liệu.

Tên báo biểu Report được tạo trong đĩa CD-ROM đính kèm là: CONNECTNATIVE.RPT.

Các trường được hiển thị bên cạnh phải của trang trong cửa sổ Field Explorer và nút Database Fields chứa danh sách các trường trong bảng đã chọn. Bạn có thể di chuyển và xóa trường từ danh sách này trên báo biểu Report của bạn.

1.2.2. Cơ sở dữ liệu quan hệ

Hầu hết các phương thức kết nối dữ liệu thông dụng nhất đều sử dụng cách kết nối ODBC để kết nối đến cơ sở dữ liệu (CSDL). Các trình điều khiển native và ODBC của Crystal Reports có thể điều khiển hầu hết các hệ CSDL quan hệ thông dụng như:

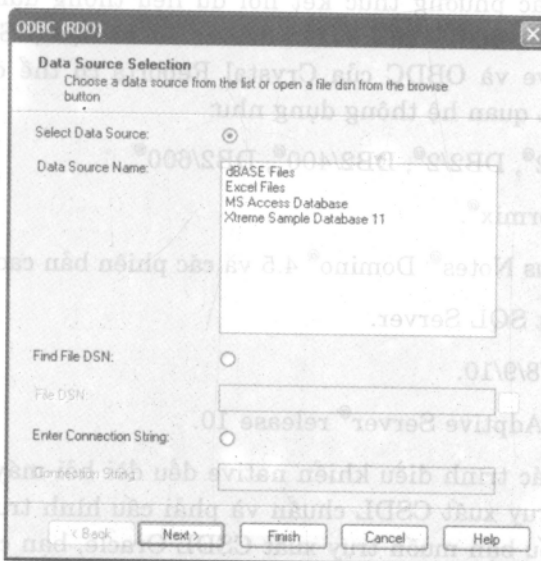
- IBM DB2®, DB2/2®, DB2/400®, DB2/600®.
- IBM Informix®.
- IBM Lotus Notes®, Domino® 4.5 và các phiên bản cao hơn như R5.
- Microsoft SQL Server.
- Oracle 7/8/9/10.
- Sybase® Adaptive Server® release 10.

Hầu hết các trình điều khiển native đều đòi hỏi máy khách (client) phải cài trình truy xuất CSDL chuẩn và phải cấu hình trước khi sử dụng chúng. Ví dụ, nếu bạn muốn truy xuất CSDL Oracle, bạn cần phải cài đặt Oracle client và cấu hình chúng trên máy của bạn. Để kiểm tra trong Database Expert xem nguồn dữ liệu native nào đang có, bạn có thể kích chọn mục More Data Sources.

Ngoài cách sử dụng trình điều khiển truy xuất native, bạn cũng có thể truy xuất các nguồn dữ liệu thông qua trình điều khiển ODBC. Để truy xuất thông qua ODBC, bạn cần phải cấu hình trình điều khiển ODBC thích hợp bằng ODBC Administrator (truy xuất thông qua Windows Control Panel) và sau đó thực hiện theo các bước sau đây:

1. Mở Crystal Reports và nhấp vào **File > New > Blank Report**. Cửa sổ **Database Expert** sẽ hiển thị.
2. Nhấp 2 lần vào biểu tượng folder để mở rộng mục **Create New Connection** và sau đó nhấp 2 lần vào mục **ODBC (RDO)**. Một danh sách các nguồn dữ liệu có sẵn xuất hiện (hình 2.3).
3. Chọn nguồn dữ liệu từ danh sách, nhập vào một số thông tin yêu cầu, sau đó nhấp vào nút **Finish**. Trong giáo trình này, phần lớn chúng ta sử dụng CSDL Xtreme làm ví dụ. CSDL này đã được cài đặt chung với Crystal Reports XI. Bạn hãy chọn nguồn dữ liệu mang tên **Xtreme Sample Database 11** và nhấp vào nút **Finish**. Một danh sách gồm

các bảng dữ liệu, view, Stored Procedures ... sẽ được hiển thị trong cửa sổ **Database Expert**.



Hình 2.3: Danh sách các nguồn dữ liệu ODBC có sẵn.

- Đánh dấu chọn bảng dữ liệu hoặc các bảng dữ liệu mà bạn muốn sử dụng trong báo biểu Report và nhấp vào nút mũi tên bên phải để di chuyển chúng qua bên danh sách đối diện. Hãy chọn bảng dữ liệu **Employee** cho ví dụ này.
- Khi hoàn thành việc chọn bảng dữ liệu, nhấp vào **OK** để trở lại cửa sổ thiết kế báo biểu Report.

Bạn có thể sử dụng các bảng dữ liệu và các trường được chọn để xây dựng báo biểu Report mong muốn.

Tên báo biểu Report của ví dụ này chứa trong thư mục Project trên CD-ROM là: Connectodbc.RPT

Ghi chú: Crystal Reports đăng nhập vào CSDL để thực thi các yêu cầu và trích rút thông tin. Khi bạn đóng một báo biểu và thoát khỏi Crystal Reports, CSDL đang kết nối vẫn còn mở chưa được đóng lại. Để tắt hay mở CSDL Server, bạn nhấp vào mục **Database > Log On or Off Server**.

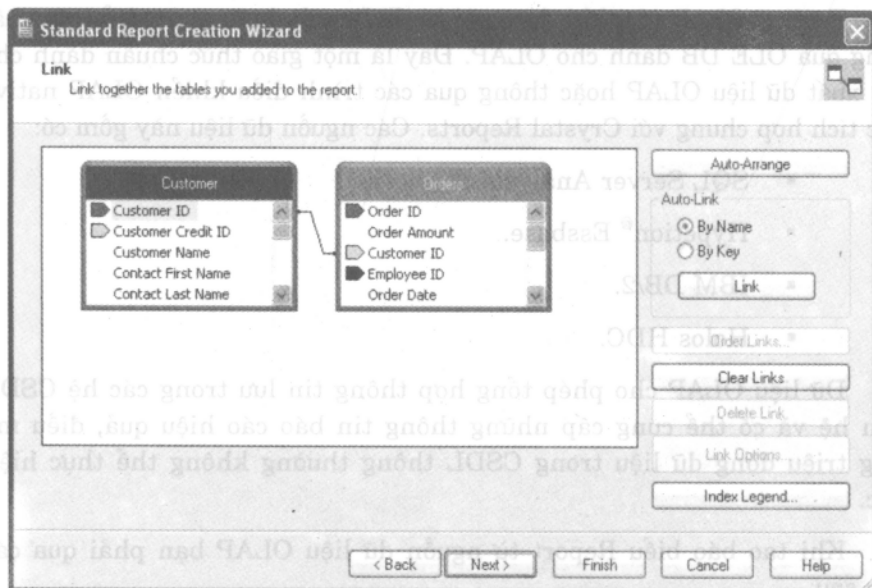
1.2.3. Làm việc với trình kết nối trực quan (Visual Linking Expert)

CSDL quan hệ (Relational Database) thường chứa các bảng dữ liệu. Những bảng dữ liệu này có thể kết hợp (join) với nhau để tạo thành một câu truy vấn phức tạp. Trong Crystal Reports, Visual Linking Expert cho phép kết hợp các bảng bằng một đường nối trực quan giữa hai trường (Field) dùng làm khóa. Nếu bạn chọn nhiều bảng dữ liệu trong Database Expert Tab, các kết nối sẽ tự tạo ra và xuất hiện trong Database Expert (hình 2.4). Hộp thoại này thường dùng để xác định quan hệ giữa các bảng dữ liệu và các View mà bạn đã chọn cho báo biểu Report.

Ghi chú: Bạn có thể tạo hộp thoại này lớn hơn bằng cách nhấp vào nút phóng lớn hộp thoại ở góc trên bên phải của hộp thoại.

Liên kết mà bạn thấy được trong hộp thoại tuân theo cú pháp kết nối SQL giữa các bảng dữ liệu khác nhau trong nguồn dữ liệu của bạn.

Crystal Reports sẽ duyệt qua các bảng, View ... mà bạn chọn cho báo biểu của mình và sẽ thử tự động liên kết chúng lại thông qua các trường khóa cũng như chỉ mục mà nó có thể tìm thấy từ tên Field, kiểu (Type) và chiều dài.



Hình 2.4: Sử dụng Visual Linking Expert xác định quan hệ giữa các bảng CSDL hoặc các file CSDL.

Nếu bạn sử dụng kiểu CSDL như Access hoặc dBase, Crystal Reports sẽ hiển thị các chỉ mục Index khác nhau có trong bảng (ví dụ, màu đỏ là chỉ mục đầu tiên, màu xanh là chỉ mục thứ hai...).

Trừ khi bạn sử dụng một cấu trúc CSDL toàn vẹn, liên kết thông minh (smart-linking) do Crystal Reports “suy đoán” thường không chính xác, bạn có thể hủy bỏ liên kết giữa các bảng hoặc các View bằng cách nhấp vào đường nối giữa hai bảng dữ liệu và nhấn nút xóa (Del) trên bàn phím của bạn. Bạn cũng có thể sử dụng nút xóa liên kết để xóa tất cả các liên kết đang tồn tại trên lược đồ kết nối.

Sau khi đã xác định đúng các kết nối bảng, bạn có thể thêm chúng vào hộp thoại bằng cách kéo một trường từ bảng này đè lên trường khóa của bảng kia để tạo nên một đường liên kết giữa 2 bảng dữ liệu. Để thay đổi kiểu kết nối giữa các bảng dữ liệu, nhấp nút chuột phải vào đường liên kết mà bạn muốn thay đổi sau đó nhấp vào mục Link Options.

Một hộp thoại sẽ hiển thị cho phép bạn chọn hoặc thay đổi kiểu kết nối (inner join, outer join và các kiểu kết nối chuẩn khác của SQL). Sau khi chỉ định kiểu kết nối bạn nhấn nút OK để trở lại Tab Links.

1.2.4. Các nguồn dữ liệu OLAP

Dữ liệu OLAP (đôi khi được gọi là dữ liệu đa hướng) có thể truy xuất thông qua OLE DB dành cho OLAP. Đây là một giao thức chuẩn dành cho truy xuất dữ liệu OLAP hoặc thông qua các trình điều khiển OLAP native được tích hợp chung với Crystal Reports. Các nguồn dữ liệu này gồm có:

- SQL Server Analysis Services.
- Hypetion® Essbase.
- IBM DB/2.
- Holos HDC.

Dữ liệu OLAP cho phép tổng hợp thông tin lưu trong các hệ CSDL quan hệ và có thể cung cấp những thông tin báo cáo hiệu quả, điều mà hàng triệu dòng dữ liệu trong CSDL thông thường không thể thực hiện được.

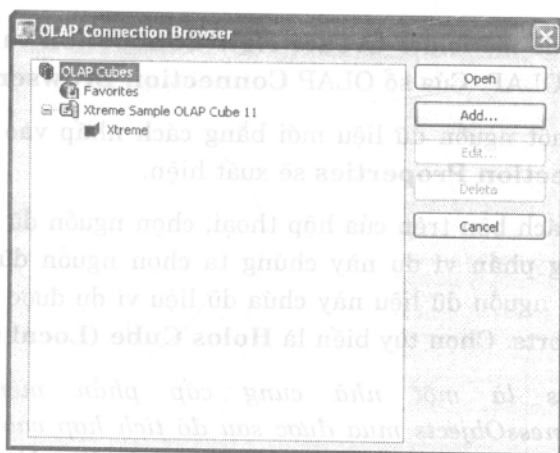
Khi tạo báo biểu Report từ nguồn dữ liệu OLAP bạn phải qua các bước sau:

1. Mở Crystal Reports và nhấp vào **File > New > Blank Report**. Cửa sổ **Database Expert** sẽ hiển thị.

2. Nhấp 2 lần để mở folder **Create New Connection** và sau đó nhấp 2 lần vào nút **OLAP**. Cửa sổ **OLAP Connection Browser** sẽ hiển thị.
3. Thêm vào một nguồn dữ liệu mới bằng cách nhấp vào nút **Add**. Hộp thoại **Connection Properties** sẽ xuất hiện.
4. Từ ô danh sách bên trên của hộp thoại, chọn nguồn dữ liệu OLAP bạn muốn. Trong phần ví dụ này chúng ta chọn nguồn dữ liệu là **Holos HDC cube**, nguồn dữ liệu này chứa dữ liệu ví dụ được cung cấp trong Crystal Reports. Chọn tùy biến là **Holos Cube (Local Client)**.

***Ghi chú:** Holos là một nhà cung cấp phần mềm OLAP mà BusinessObjects mua được sau đó tích hợp vào sản phẩm của mình. Holos HDC Cubes là nguồn dữ liệu lý tưởng cho chức năng OLAP trong Crystal Reports, do nó không phụ thuộc vào chương trình OLAP Server.*

5. Trong ô TextBox hãy nhập vào tên tiêu đề cho nguồn OLAP của bạn. Trong ví dụ này ta nhập vào chuỗi **Xtreme Sample OLAP Cube 11**.
6. Sử dụng nút **Browse** để chọn nguồn dữ liệu OLAP. Ví dụ, khối dữ liệu **cube** trong ví dụ này nằm trong thư mục:
x:\Program Files\Business Objects\Crystal Reports 11\Samples\En\Databases\Olap Data\Xtreme.hdc
(Trong đó, x là thư mục mà bạn cài chương trình Crystal Reports trên máy tính).
7. Kế đến, nhấp vào nút **Test Connection** để kiểm tra kết nối. Một hộp thông báo sẽ xuất hiện thông báo kết quả kết nối thành công.
8. Để hoàn tất thiết lập, bạn nhấp vào nút **OK** để trở về hộp thoại **OLAP Connection Browser** và ta sẽ có được kết quả kết nối OLAP như hình 2.5.
9. Mở rộng nguồn dữ liệu bạn mới tạo và đánh dấu chọn **cube** mà bạn muốn tạo báo biểu Report. Trong ví dụ này, mở rộng nguồn dữ liệu, chọn **Xtreme** và nhấp vào **Open**. Bạn sẽ trở lại **Database Expert** và tất cả giá trị **Cube** chứa trong nguồn dữ liệu sẽ xuất hiện bên dưới tên thư mục OLAP.
10. Đánh dấu chọn **Cube** mà bạn muốn sử dụng trong báo biểu Report và nhấp vào nút mũi tên chỉ về bên phải để di chuyển chúng từ hộp danh sách bên này sang hộp danh sách bên kia. Trong phần này ta chọn **Cube Customer**.



Hình 2.5: OLAP Connection Browser hiển thị kết nối bạn mới tạo.

Lưu ý: Nếu CSDL quan hệ lưu dữ liệu ở dạng bảng (table) thì OLAP lưu dữ liệu dạng khối 3 chiều (Cube).

11. Khi hoàn tất việc chọn lựa bảng dữ liệu, nhấp nút OK để trở về cửa sổ thiết kế báo biểu Report.

Một danh sách giá trị các trường sẽ xuất hiện trong cửa sổ **Field Explorer** nằm bên phía phải cửa sổ thiết kế báo biểu Report. Bạn có thể kéo và thả các trường Field vào trong báo biểu Report mà bạn muốn tạo. Nếu các trường Field trông có vẻ không gọn gàng (ví dụ: “Monthly, Level 0”), đừng quan tâm đến điều đó vào lúc này.

Bạn có thể tham khảo mẫu báo biểu Report của ví dụ này trên đĩa CD-ROM kèm theo giáo trình, nó có tên là Connectolap.PRT.

Ghi chú: Nên nhớ các bước của thủ tục này có thể khác với những nguồn dữ liệu OLAP khác nhau mà Crystal Reports hỗ trợ. Ví dụ, để thiết lập kết nối với nguồn Essbase, bạn cần phải nhập vào tên Server và username/password, những thông tin này là cần thiết đối với phần mềm Essbase Client cài đặt và thiết lập trên máy tính bạn.

1.2.5. Các nguồn dữ liệu doanh nghiệp

Các hệ thống quản trị doanh nghiệp ERP tuy chạy ở lớp trên CSDL quan hệ nhưng chúng có các cấu trúc dữ liệu và các qui tắc nghiệp vụ riêng. Crystal Reports cung cấp khá nhiều các trình kiểu điều khiển dữ liệu cho những hệ thống ERP lớn như:

- SAP.
- BeanTM.
- PeopleSoft[®].
- PeopleSoft Enterprise One (trước là J.D. Edwards).

Ghi chú: Một vài trình điều khiển có thể được bán riêng không kèm theo Crystal Reports XI.

Để thiết lập các trình điều khiển và truy xuất thông tin, bạn phải nhờ đến sự giúp đỡ của nhà quản trị ứng dụng, bởi vì việc thiết lập không dễ chút nào đối với một người chưa rành hệ thống, nó sẽ làm bạn nản chí. Một số kỹ thuật trên trang web BusinessObjects (http://support.businessobjects.com/library/docfiles/cps10/docs_en.asp) sẽ cung cấp các kỹ thuật nền tảng cần thiết nhất cho người bắt đầu học. Thường nó sẽ yêu cầu cấu hình và phần mềm phía máy khách (client software) hoặc phiên bản OEM hay một số thành phần thêm vào cho Crystal Reports để sử dụng các trình điều khiển đó.

1.2.6. Các nguồn dữ liệu khác

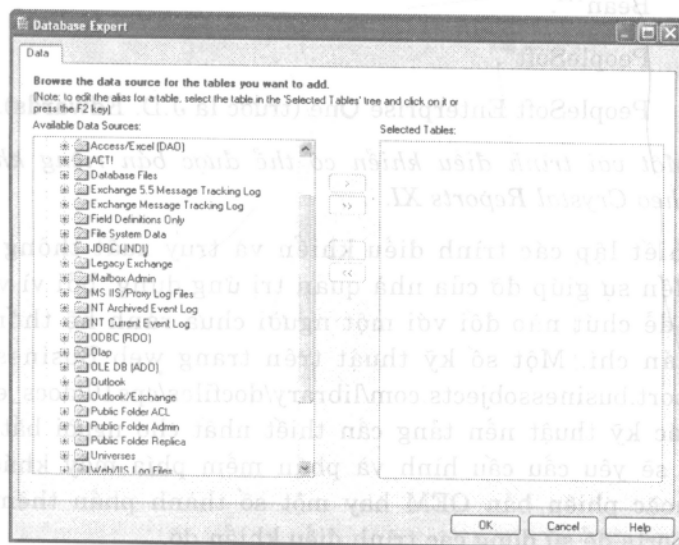
Crystal Reports cũng kèm theo một số trình điều khiển cho các nguồn dữ liệu không chuẩn như Act![®], Microsoft Exchange và Microsoft Logs. Hầu hết các nguồn dữ liệu này đều có thiết lập cấu hình đặc thù và không hợp với cách cấu hình của những nguồn dữ liệu chuẩn có thể truy xuất thông qua trình điều khiển native (native) hoặc ODBC. Một trong các kiểu định dạng dữ liệu đó là:

- Act! 3.0.
- Microsoft Exchange.
- Microsoft Outlook.
- Lotus Domino.
- Web/II's Log Files.

Tất cả các kiểu dữ liệu trên đều có sẵn trong Database Expert, bạn có thể tìm thấy nó trong phần chính hoặc trong mục More Data Sources như hình 2.6 và hầu hết đều yêu cầu cung cấp thêm thông tin kết nối (như tên File, vị trí, tên Server, v.v....).

Ghi chú: Để biết được danh sách thông tin đầy đủ của các nguồn dữ liệu này bạn hãy xem trong hướng dẫn của Crystal Reports bằng

cách nhấp vào *Help > Crystal Reports Help* trong cửa sổ thiết kế báo biểu Report.



Hình 2.6: Các nguồn dữ liệu khác có thể sử dụng trong Crystal Reports.

2. TẠO BÁO BIỂU REPORT VỚI STANDARD REPORT WIZARD

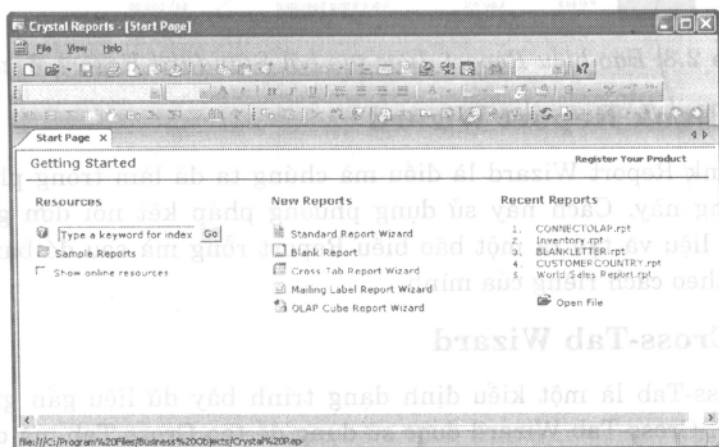
Cho đến lúc này, chúng ta đã lướt qua các bước đầu tiên trong thiết kế báo biểu Report đó là tạo kết nối đến dữ liệu và tạo một báo biểu Report rỗng. Khi bạn đã tạo xong kết nối, chọn các bảng dữ liệu, view hay Stored Procedure và từ đó bạn có thể trích dữ liệu tạo nên báo biểu Report cho mình. Trong trường dữ liệu sẽ xuất hiện cửa sổ Field Explorer, bạn có thể nắm kéo thả chúng vào báo biểu Report của mình.

Mặc dù đây là phương thức trực tiếp để tạo báo biểu Report, nhưng với người mới bắt đầu sử dụng, có một cách khác dễ dàng hơn. Crystal Reports có khá nhiều trình tạo báo biểu tự động theo các bước Wizards, kiểu tạo này sẽ giúp bạn từng bước xử lý thiết kế báo biểu Report, từng bước từ lúc bắt đầu kết nối dữ liệu cho đến khi hoàn tất báo biểu Report. Hầu hết các nhà phát triển đều sử dụng cách này để tạo báo biểu Report và sau đó sẽ chỉnh sửa lại theo ý mình.

Ghi chú: Nên nhớ mọi thứ mà báo biểu Report Design Wizard sử dụng cũng có thể hoàn tất thông qua sử dụng các mục chọn Menu,

các dòng lệnh và các biểu tượng ngay chính trong môi trường thiết kế báo biểu Report Designer. Nếu bạn thích cách đầu hơn, bạn có thể sử dụng kỹ thuật phác thảo cho báo biểu Report trước và thêm tất cả các thành phần của chúng vào báo biểu Report. Tuy nhiên, sử dụng Wizards sẽ khởi tạo báo biểu Report dễ dàng hơn.

Khi bạn lần đầu tiên mở Crystal Reports, bạn sẽ thấy hiện diện trang chủ của ứng dụng như hình 2.7.



Hình 2.7: Crystal Reports Start Page.

Trang này cung cấp một danh sách tài nguyên mà bạn có thể dùng để bắt đầu sử dụng Crystal Reports. Nó cũng cung cấp một danh sách các báo biểu Report Wizards mà bạn có thể tương tác bằng cách nhấp vào mục File > New. Danh sách của các Wizards này sẽ được đề cập trong phần sau, bao gồm mô tả vắn tắt nội dung và cách sử dụng chúng.

2.1. Standard Report Wizard

Standard Report Wizard rất thường được sử dụng. Nó cho phép bạn tạo báo biểu Report dạng liệt kê, tạo nhóm (Group), sắp xếp (sort), gom tổng (summary) dữ liệu cần báo cáo...

Bạn cũng có thể tạo các báo biểu Report dạng phân tích (Analysis) bằng cách sử dụng Standard Report Wizard (ví dụ như phân tích xem 10 khách hàng mua nhiều nhất là ai), thêm hình minh họa và áp dụng các kiểu xác định trước cho báo biểu Report của bạn. Một ví dụ minh họa cho việc sử dụng Standard Report Wizard để tạo báo biểu Report được thể hiện ở hình 2.8.

BUSINESS OBJECTS		Employee Listing	
Last Name	First Name	Position	Salary
Davolio	Nancy	Sales Representative	\$40,000.00
Fuller	Andrew	Vice President, Sales	\$90,000.00
Leverling	Jane	Sales Representative	\$33,000.00
Peacock	Margaret	Sales Representative	\$35,000.00
Buchanan	Sтивен	Sales Manager	\$50,000.00
Suyama	Michael	Sales Representative	\$30,000.00
King	Robert	Sales Representative	\$37,000.00
Callahan	Laura	Inside Sales Coordinator	\$45,000.00
Doddsworth	Anne	Sales Representative	\$35,000.00
Hallstrom	Albert	Business Manager	\$60,000.00
Smith	Tim	Mail Clerk	\$18,000.00
Patterson	Caroline	Receptionist	\$25,000.00
Britt	Justin	Marketing Director	\$75,000.00
Martin	Xavier	Marketing Associate	\$50,000.00
Pereira	Laurent	Advertising Specialist	\$45,000.00

Hình 2.8: Báo biểu Report được tạo bởi Standard Report Wizard.

2.2. Blank Report Wizard

Blank Report Wizard là điều mà chúng ta đã làm trong phần trên của chương này. Cách này sử dụng phương pháp kết nối đơn giản đến nguồn dữ liệu và tạo ra một báo biểu Report rộng mà sau đó bạn có thể tùy biến theo cách riêng của mình.

2.3. Cross-Tab Wizard

Cross-Tab là một kiểu định dạng trình bày dữ liệu gần giống với bảng tính. Cross-Tab Wizard được sử dụng để tạo Cross-Tab mà các dòng và các cột có thể kết hợp với nhau để tạo nên dạng tổng hợp (Summary) hoặc phân cấp. Ngoài dòng và cột, Cross-Tab cũng tính tổng các Field và cho phép bạn đọc thông tin báo biểu nhanh chóng từ dạng Cross-Tab phân cấp và nhóm (hình 2.9)

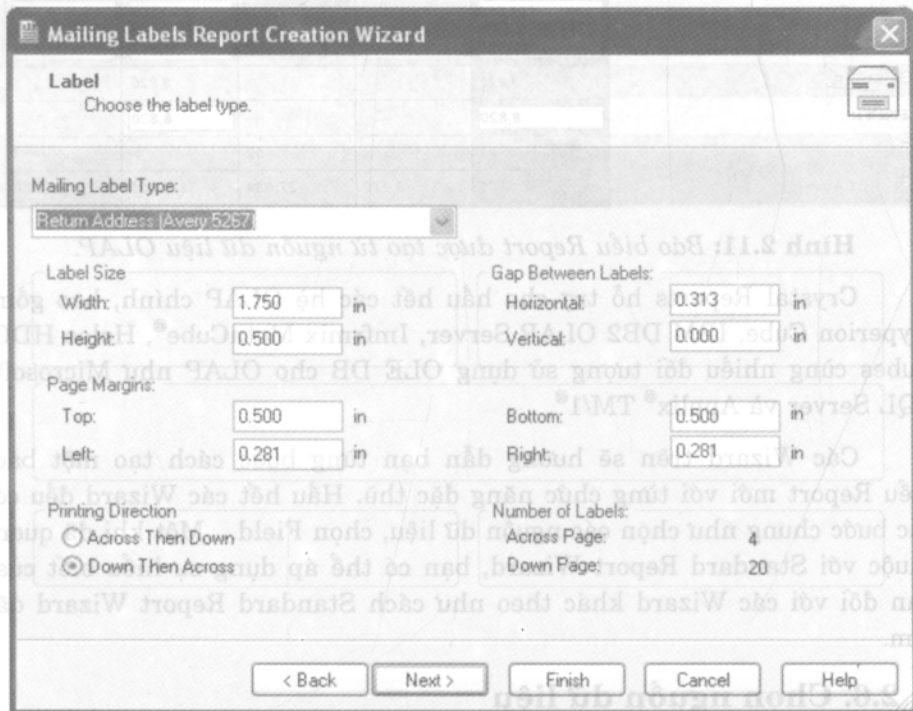
Quarterly Sales by Country

		1/2000	10/2000	1/2001	4/2001	7/2001	10/2001
USA	AL	\$1,067.50	\$4,717.50	\$14,060.26	\$47,611.07	\$38,824.45	\$32,113.87
	AR	\$0.00	\$0.00	\$0.00	\$8,919.55	\$0.00	\$0.00
	AZ	\$0.00	\$0.00	\$0.00	\$91.40	\$0.00	\$0.00
	CA	\$5,526.63	\$2,860.65	\$64,680.89	\$47,630.29	\$40,181.21	\$55,095.34
	CO	\$0.00	\$0.00	\$0.00	\$161.70	\$0.00	\$0.00
	CT	\$0.00	\$0.00	\$0.00	\$43.50	\$0.00	\$0.00
	DC	\$0.00	\$0.00	\$0.00	\$13.50	\$0.00	\$0.00
	DE	\$0.00	\$0.00	\$0.00	\$52.50	\$0.00	\$0.00
	FL	\$1,968.18	\$1,318.50	\$14,606.71	\$20,369.23	\$30,877.67	\$10,357.65
	GA	\$0.00	\$0.00	\$0.00	\$17.50	\$0.00	\$0.00
	IA	\$11,363.10	\$5,893.09	\$12,324.12	\$25,836.51	\$22,654.39	\$25,926.62

Hình 2.9: Ví dụ về báo biểu Report Cross-Tab.

2.4. Mailing Label Wizard (tạo nhãn thư tự động)

Crystal Reports có thể tạo ra các nhãn từ nguồn dữ liệu một cách dễ dàng và nhanh chóng dựa vào các nhãn chuẩn được xác định trước. Sử dụng Mailing Label Wizard bạn có thể tạo nhãn nhanh với các kích thước và các định dạng khác nhau mà không cần xác định khuôn khổ, khoảng cách... Nếu bạn cần tạo một nhãn với kích thước hay định dạng không có trong danh sách các mẫu nhãn chuẩn, bạn có thể nhập vào kích thước và khoảng cách để tạo nhãn tùy biến (hình 2.10).



Hình 2.10: Thiết lập tùy biến kích thước nhãn.

2.5. Online Analytical Processing (OLAP) Wizard

Online Analytical Processing (OLAP) Wizard có thể dùng để tạo các báo biểu Report từ các nguồn dữ liệu OLAP. Khác với CSDL quan hệ chứa dữ liệu trong các bảng dữ liệu ở dạng mảng hai chiều, cấu trúc OLAP biểu diễn dữ liệu theo mảng đa chiều. Thay cho các dòng và cột (là mảng hai chiều), cấu trúc OLAP là một mảng đa chiều giúp cho người kinh doanh và nhà quản lý phân tích dữ liệu có thể trích rút thông tin tổng hợp (hình 2.11).

Sales by Product/Country

Year Total					
All Products					
		Active Outdoors Crochet Glove	Active Outdoors Lycra Glove	Discount	Endorphin
USA	4,066,527	106,125	137,839	1,080,018	183,023
AL	259,943	12,276	855	78,257	27,407
Benny - The Spokes Person	35,525	1,570			2,780
Psycho-Cycle	121,566		855	50,903	3,147
The Great Bike Shop	102,852	10,706		27,353	21,479
AR	8,820			8,820	
Bikefest	8,820			8,820	
AZ	51		50		
CA	401,765	8,170	27,024	125,936	16,527

Hình 2.11: Báo biểu Report được tạo từ nguồn dữ liệu OLAP.

Crystal Reports hỗ trợ cho hầu hết các hệ OLAP chính, bao gồm Hyperion Cube, IBM DB2 OLAP Server, Imfrmix MetaCube®, Holos HDC Cubes cùng nhiều đối tượng sử dụng OLE DB cho OLAP như Microsoft SQL Server và Applix® TM/1®.

Các Wizard trên sẽ hướng dẫn bạn từng bước cách tạo một báo biểu Report mới với từng chức năng đặc thù. Hầu hết các Wizard đều có các bước chung như chọn các nguồn dữ liệu, chọn Field... Một khi đã quen thuộc với Standard Report Wizard, bạn có thể áp dụng sự hiểu biết của bạn đối với các Wizard khác theo như cách Standard Report Wizard đã làm.

2.6. Chọn nguồn dữ liệu

Bước đầu tiên khi tạo một báo biểu Report bằng Standard Report Wizard là chọn nguồn dữ liệu mà báo biểu muốn sử dụng. Wizard sẽ hiển thị danh sách tất cả các nguồn dữ liệu mà nó hỗ trợ.

- **Current Connections:** Hiển thị các nguồn dữ liệu đang kết nối.
- **Repository:** Nếu người quản trị của bạn cấu hình nguồn dữ liệu trong BusinessObjects Repository thì nó sẽ xuất hiện ở đây.
- **Favorites:** Nếu bạn đã quen và yêu thích Internet Explorer, tùy biến này sẽ khá quen thuộc, nó hoạt động theo cách gần giống

như vậy, nhưng thay vì lưu những địa chỉ trang Web thường xuyên ghé thăm, nó cho phép bạn lưu thường xuyên các kết nối dữ liệu hay sử dụng.

- **History:** Phần này cho phép tự động lưu lại các kết nối bạn vừa sử dụng gần đây nhất.
- **Create a New Connection:** Lệnh này được dùng để tạo một kết nối mới đến nguồn dữ liệu của bạn.

Bạn đã xem qua một số tùy chọn này từ đầu chương khi tạo báo biểu Report rỗng đơn giản từ một nguồn dữ liệu (nếu bạn mở rộng nút Create New Connect, bạn sẽ thấy một số các nguồn dữ liệu khác nhau trong đó và bạn có thể sử dụng chúng để tạo báo biểu Report của bạn).

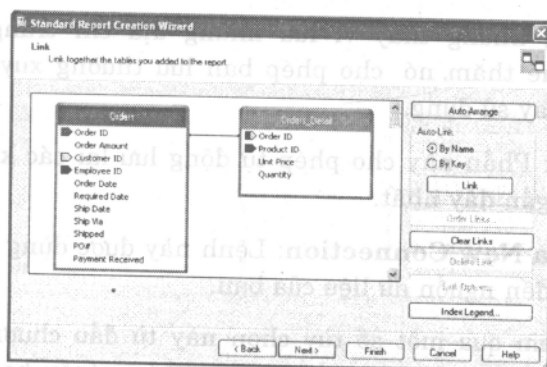
Ví dụ, để tạo một báo biểu Report từ nguồn dữ liệu kết nối ODBC, bạn sẽ phải mở rộng nút ODBC Data Source. Một hộp thoại thứ hai sẽ xuất hiện yêu cầu bạn chọn nguồn dữ liệu mình muốn dùng. Khi đã chọn nguồn dữ liệu, một nút mới sẽ được tạo ra bên dưới nút ODBC hiển thị nguồn dữ liệu bạn vừa chọn và tất cả các bảng dữ liệu, View, Stored Procedure sẵn sàng để bạn sử dụng.

Để thêm một bảng dữ liệu vào báo biểu Report, mở rộng nút bảng Tables và nhấp 2 lần vào danh sách bảng cần chọn để di chuyển bảng được chọn qua phải hoặc bạn cũng có thể đánh dấu chọn bảng dữ liệu và sử dụng nút mũi tên để di chuyển chúng sang danh sách bên phải.

2.7. Liên kết các bảng CSDL hoặc các File

Nếu bạn chọn một hoặc nhiều bảng dữ liệu cho báo biểu Report của mình, Standard Report Wizard sẽ thêm vào một bước trong Wizard với tiêu đề là Link (hình 2.12). Lời khuyên tốt nhất là tìm một mô hình quan hệ thực thể có thể mô phỏng được sự quan hệ giữa các bảng trong CSDL và sử dụng quan hệ này như ví dụ ở đây. Bạn tạo quan hệ giữa các bảng trong Crystal Reports bằng cách vẽ đường liên kết giữa các trường trong bảng dữ liệu với nhau.

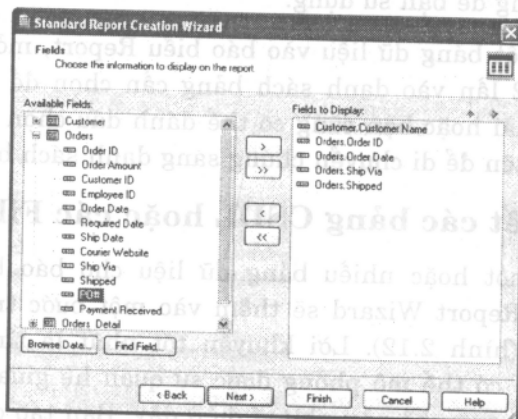
Mặc định Crystal Reports sẽ tự thực hiện các liên kết hay kết nối cho dữ liệu của bạn trên các tên trường hay khóa định nghĩa trong các bảng dữ liệu. Việc liên kết các bảng được đề cập chi tiết trong phần cuối của chương này. Bây giờ, cần chọn một số trường (Field) sẽ xuất hiện trong báo biểu Report của chúng ta.



Hình 2.12: Liên kết Tab giúp bạn kết nối hai hay nhiều bảng dữ liệu lại với nhau.

2.8. Chọn trường dữ liệu

Trang Field của Standard Report Wizard (hình 2.13) được chia thành hai phần. Khung bên trái của hộp thoại là tất cả danh sách của các trường có thể chọn để thêm vào báo biểu Report của bạn, nhóm (Group) bên dưới bảng dữ liệu của chúng.



Hình 2.13: Sử dụng trang Field để chọn các trường cho báo biểu Report.

Để thêm Field vào báo biểu Report, bạn cần di chuyển Field từ khung bên trái qua khung bên phải. Bạn thực hiện việc này bằng cách nhấn 2 lần vào tên Field hay bằng cách đánh dấu chọn và nhấn vào nút Add. Một số nút khác được dùng giúp bạn thêm hay loại bỏ tất cả các trường đang được chọn.

Ghi chú: Để chọn nhiều trường Field cùng lúc, nhấn và giữ phím **Ctrl** trong khi chọn.

Nếu bạn không xác định được Field hay nội dung của Field, bạn có thể nhấp vào nút **Browse Data** để hiển thị nội dung Field mẫu. Lưu ý rằng, mẫu trả về không chỉ là một phần của bảng dữ liệu (khoảng 200 mẫu tin).

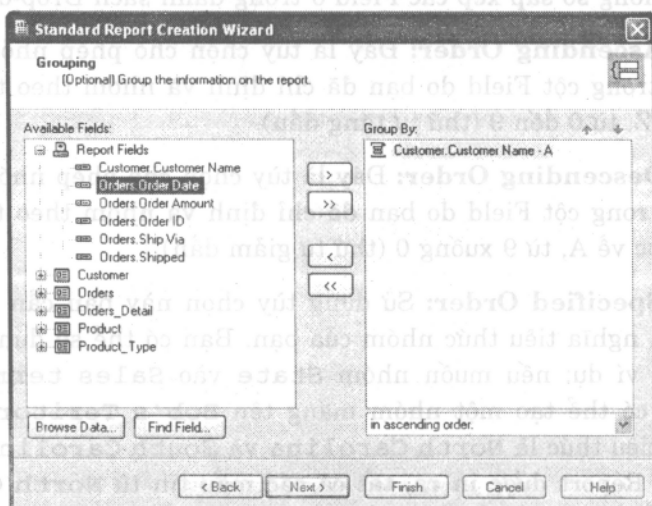
Một chức năng khác của trang Field là nút tìm kiếm **Find Field**, phía dưới góc bên trái. Nút này cho phép bạn tìm kiếm các Field trong bảng dữ liệu theo một điều kiện nào đó.

Do đây mới chỉ là bước bắt đầu học thiết kế báo biểu Report, chúng ta không thảo luận nhiều về nút Formula dùng thiết lập công thức tính toán. Nếu bạn không thể đợi mà muốn học ngay cách sử dụng công thức, bạn có thể tìm hiểu ở chương 5 của giáo trình này.

Trong lúc thêm các trường vào báo biểu Report, để thay đổi thứ tự của các Field, bạn có thể sử dụng mũi tên lên hoặc xuống ở phía góc trên bên phải hộp thoại để di chuyển các Field lên xuống theo ý mình. Tại đây bạn cũng có thể thay đổi tiêu đề của các Field.

2.9. Nhóm và sắp xếp

Bước tiếp theo trong tạo báo biểu Report bằng Standard Report Wizard là chọn cách sắp xếp và nhóm các đối tượng sử dụng trong báo biểu Report của bạn bằng cách sử dụng mục Grouping (hình 2.14). Muốn nhóm một Field, bạn phải di chuyển Field muốn nhóm từ danh sách bên trái hộp thoại sang bên phải hộp thoại.



Hình 2.14: Tùy chọn gom nhóm.

Bằng cách chỉ định một trường trên Group Tab, bạn có thể thêm điều khiển ngắt hay nhóm vào báo biểu Report của mình. Ví dụ, nếu bạn nhóm Field mang tên **State**, báo biểu Report của bạn sẽ in tất cả các mẫu tin trên mỗi **State** khác nhau với ngắt phân cách ở giữa mỗi **State** (hình 2.15).

<u>Region</u>	<u>Customer Name</u>	<u>Address</u>	<u>City</u>
<u>Abu Dhabi</u>			
Abu Dhabi	UAE Cycle	Post Box 278	Abu Dhabi
<u>Abu Dhabi</u>			
AL			
AL	The Great Bike Shop	1822 Beach Crescent	Huntsville
AL	Benny - The Spokes Person	1020 Oak Way	Huntsville
AL	Psycho-Cycle	Customer Customer Name (String)	Huntsville
<u>Alsace</u>			
Alsace	Sports Alsace	11, rue Clemenceau	Strasbourg
Alsace	Mulhouse Vélos	4 avenue de la Liberation	Mulhouse
<u>Ankara</u>			
Ankara	Ankara Bicycle Company	P.O. Box 2121	Ankara
<u>Aquitaine</u>			
Aquitaine	Bordeaux Sports	120, avenue des Pins	Bordeaux
Aquitaine	Lourdes Sports	54, avenue Pierre	Lourdes
<u>Aquitaine</u>			

Hình 2.15: Ví dụ về báo biểu Report nhóm bởi Field State.

Các thông số sắp xếp các Field ở trong danh sách Drop-down là:

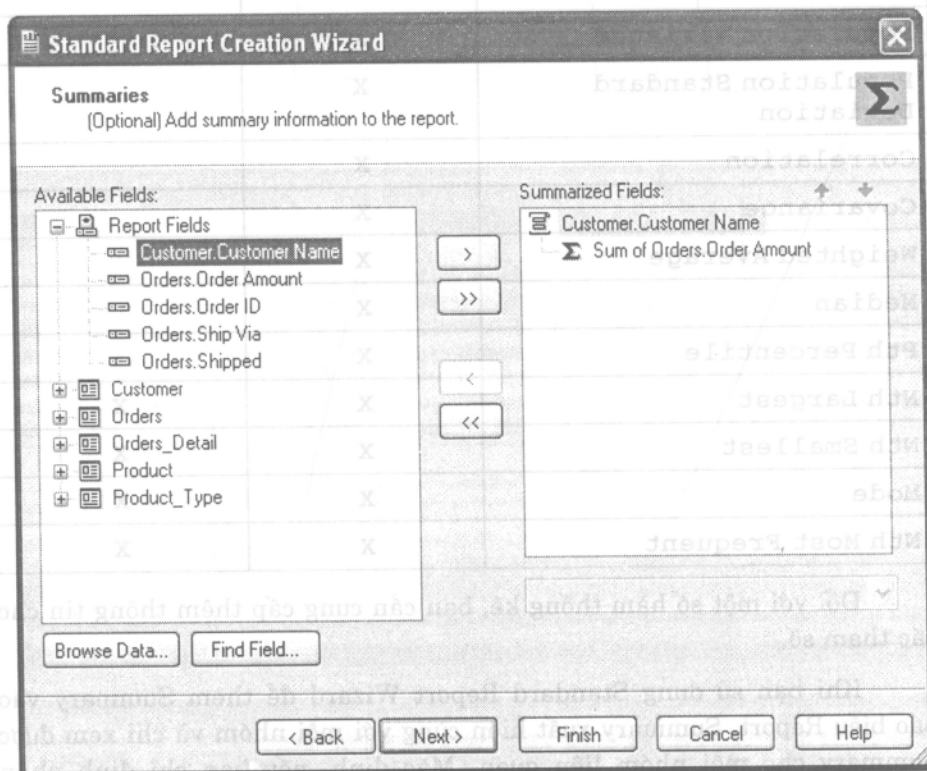
- **In Ascending Order:** Đây là tùy chọn cho phép nhóm các mẫu tin trong cột Field do bạn đã chỉ định và nhóm theo thứ tự từ A đến Z, từ 0 đến 9 (thứ tự tăng dần).
- **In Descending Order:** Đây là tùy chọn cho phép nhóm các mẫu tin trong cột Field do bạn đã chỉ định và nhóm theo thứ tự từ Z ngược về A, từ 9 xuống 0 (thứ tự giảm dần).
- **In Specified Order:** Sử dụng tùy chọn này bạn cần đặt tên và định nghĩa tiêu thức nhóm của bạn. Bạn có thể sử dụng tùy chọn này, ví dụ: nếu muốn nhóm State vào Sales territories, bạn có thể tạo một nhóm mang tên Bob's Territory và thiết lập tiêu thức là North Carolina và South Carolina. Khi báo biểu Report được in ra, tất cả các mẫu tin từ North Carolina và South Carolina sẽ được nhóm lại dưới nhóm mang tên Bob's Territory.

- **In Original Order:** Nếu CSDL của bạn đã thực thi việc sắp xếp trên dữ liệu trước đó, tùy chọn này giữ nguyên thứ tự các mẫu tin không sắp xếp.

Ghi chú: Trong bước Grouping này của Wizard, nút chức năng **Browse Data** hỗ trợ bạn tìm một Field trong các nhóm dữ liệu.

2.10. Thêm phần tổng gộp

Tùy chọn Summary (hình 2.16) được sử dụng để Crystal Reports thêm Field tổng gộp Summary vào báo biểu Report của bạn. Summary chỉ xuất hiện khi bạn thêm nhóm sử dụng Wizard. Crystal Reports cung cấp các Field tính tổng gộp để bạn không cần phải tạo công thức mỗi khi bạn muốn tính tổng (sum), tính giá trị trung bình (Average)...



Hình 2.16: Thêm Field Summary vào báo biểu Report của bạn.

Bạn cần thêm một Field Summary vào báo biểu Report của bạn, tuy nhiên các kiểu Summary sẽ khác nhau đối với mỗi loại Field khác nhau, hãy xem bảng mô tả sau:

Kiểu	Với trường kiểu số	Với các kiểu khác
Sum	X	
Average	X	
Maximum	X	X
Minimum	X	X
Count	X	X
Distinct Count	X	X
Sample Variance	X	
Sample Standard Deviation	X	
Population Variance	X	
Population Standard Deviation	X	
Correlation	X	
Covariance	X	
Weighted Average	X	
Median	X	
Pth Percentile	X	
Nth Largest	X	X
Nth Smallest	X	X
Mode	X	X
Nth Most Frequent	X	X

Đối với một số hàm thống kê, bạn cần cung cấp thêm thông tin cho các tham số.

Khi bạn sử dụng Standard Report Wizard để thêm Summary vào báo biểu Report, Summary xuất hiện cùng với mỗi nhóm và chỉ xem được Summary cho mỗi nhóm liên quan. Mặc định, nếu bạn chỉ định phân nhóm cho báo biểu Report của mình, Crystal Reports sẽ thêm vào mục tính tổng gộp cho tất cả các Field bạn đã chọn.

2.11. Sắp xếp nhóm

Sắp xếp nhóm (hình 2.17) (đôi khi còn gọi là Top N, Bottom N) là chức năng phân tích mạnh cho phép bạn sắp thứ tự dữ liệu trên tổng con (Subtotal) hoặc tổng cuối (Summary) và là một bước tùy chọn có trong hầu hết các Report Wizard.

Ví dụ, nếu bạn có báo biểu Report tính tổng của mỗi khách hàng đã bán trong năm, bạn cần sử dụng sắp xếp nhóm để xác định vị trí 10 hay 20 khách hàng bán được nhiều nhất (tương tự như vậy, bạn cũng tìm được 10 khách hàng thấp nhất).

Ngoài ra, đánh giá Top N và Bottom N còn cho phép sắp xếp các khách hàng của bạn, địa danh theo thứ bậc từ cao xuống thấp.

Khi làm việc với nhóm sắp xếp và Report Wizard, bạn cần lưu ý 2 điều:

- Chú ý đầu tiên là đối với phân tích Top N, bạn cần có một nhóm để thêm vào báo biểu Report của mình và tính tổng các Field trong nhóm đó.
- Chú ý thứ hai là đối với phân tích nhóm sắp xếp thì cần chấp nhận đa nhóm. Ví dụ, bạn có thể muốn xem danh sách 10 khách hàng cao nhất và 5 sản phẩm bán chạy nhất.

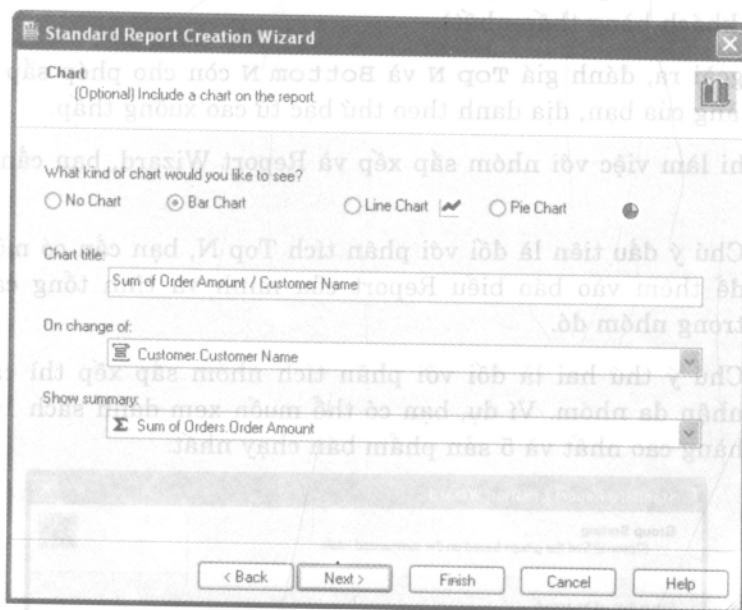


Hình 2.17: Nhóm sắp xếp và phân tích Top N dùng để đánh giá dữ liệu báo biểu Report.

2.12. Thêm vào biểu đồ Chart

Crystal Reports sử dụng sức mạnh thiết kế đồ họa của hãng sản xuất thứ ba để cung cấp đồ họa Graphics và biểu đồ Charts (hình 2.18). Một trong các kiểu Chart được sử dụng trong Wizard là:

- Bar (thanh).
- Line (đường thẳng).
- Pie (bánh).

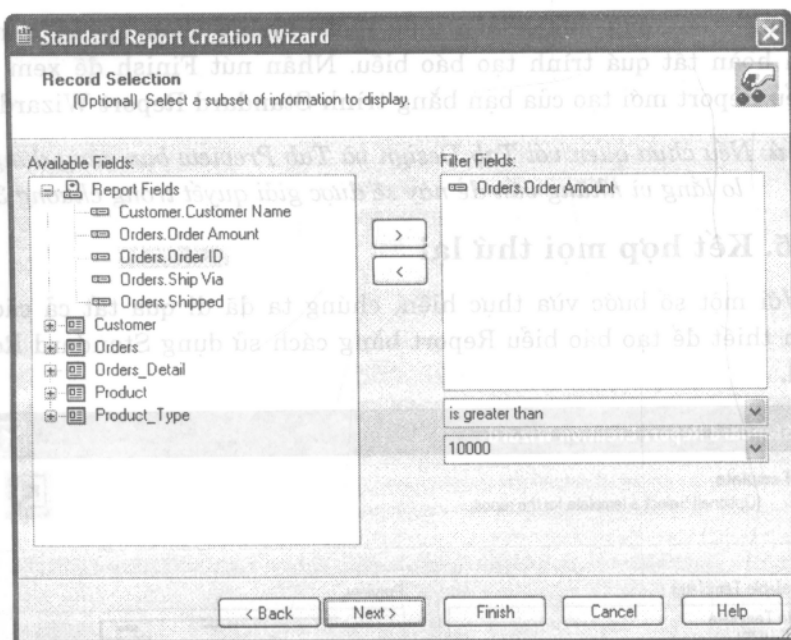


Hình 2.18: Charts và Graphics có thể sử dụng trong Crystal Reports.

Bạn cũng có thể tùy biến để tạo ra kiểu biểu đồ Chart đặc thù cho riêng mình. Ngoài ra với các kiểu đồ họa Graphic, bạn có thể chọn nguồn dữ liệu cho ảnh đồ họa. Chức năng biểu đồ Wizard Chart chủ yếu dựa trên nhóm mà bạn đưa vào và một số Field tổng gộp. Chức năng biểu đồ sẽ được đề cập chi tiết trong chương 7.

2.13. Tuyển chọn mẫu tin cho báo biểu

Một trong những bước cuối cùng của quá trình tạo báo biểu Report sử dụng Standard Report Wizard là thiết lập chọn lọc mẫu tin cho báo biểu Report của bạn bằng cách sử dụng mục Report Selection Option (hình 2.19).



Hình 2.19: Chọn mẫu tin chọn lọc cho báo biểu Report của bạn.

Record Selection khá quan trọng do nó giúp bạn loại bỏ những dòng mẫu tin không cần đến trong bảng dữ liệu. Bạn sử dụng Record Selection để lọc lại thông tin. Ví dụ, nếu bạn đã tạo xong báo biểu Report bán hàng hằng ngày, bạn có thể chỉ cần lấy về những mẫu tin bán hàng trong một ngày cụ thể nào đó; hay chỉ trả về thông tin của một khách hàng thay vì cả triệu khách hàng đang có.

Để sử dụng bước này trong Wizard, bạn cần chỉ rõ Field dùng để lọc sau đó thiết lập các tiêu chuẩn lọc cho mẫu tin dựa trên Field được chọn. Nếu bạn tạo báo biểu Report từ một CSDL tương đối nhỏ, bạn có thể bỏ qua bước này để chuyển đến bước chọn khuôn mẫu (Selecting Template) và xem trước báo biểu Report.

2.14. Chọn khuôn mẫu (Template)

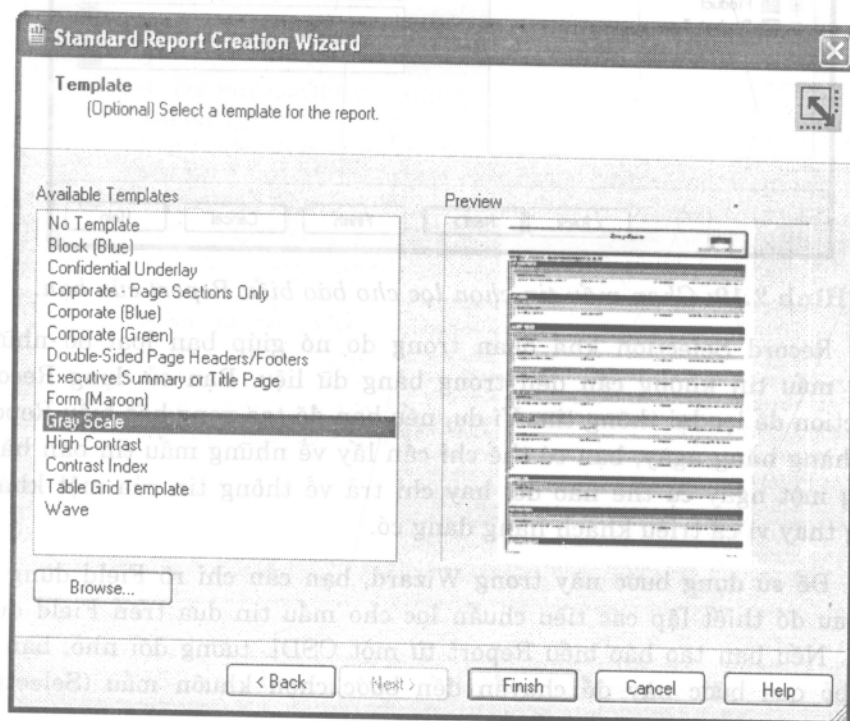
Bước cuối cùng trong Standard Report Wizard là áp dụng khuôn mẫu cho báo biểu Report của bạn (hình 2.20). Khuôn mẫu là một tập các thuộc tính định dạng được áp dụng cho báo biểu Report của bạn. Bạn có thể chọn các kiểu khuôn mẫu được định nghĩa trước và xem ngay kết quả trong khung Preview bên phải. Bạn cũng có thể chọn khuôn mẫu không có trong danh sách bằng cách nhấp vào nút Browse và chọn file Template.

Khi chọn xong một khuôn mẫu (hoặc không chọn gì cả), bạn gần như đã hoàn tất quá trình tạo báo biểu. Nhấn nút Finish để xem trước báo biểu Report mới tạo của bạn bằng trình Standard Report Wizard.

Ghi chú: Nếu chưa quen với Tab Design và Tab Preview bạn cũng đừng quá lo lắng vì những vấn đề này sẽ được giải quyết trong chương 3.

2.15. Kết hợp mọi thứ lại

Với một số bước vừa thực hiện, chúng ta đã đi qua tất cả các yêu cầu cần thiết để tạo báo biểu Report bằng cách sử dụng Standard Report Wizard.

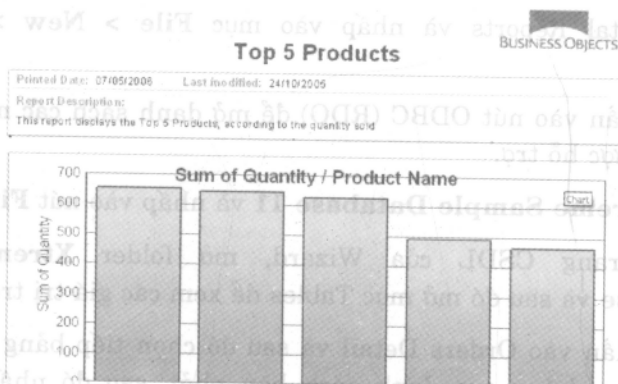


Hình 2.20: Chọn Template cho báo biểu Report.

Trong phần này chúng ta sẽ thực hiện tạo một báo biểu đầy đủ và nghiêm chỉnh. Báo biểu Report này sẽ hiển thị nhóm 5 sản phẩm bán ra nhiều nhất và sẽ bao gồm hình ảnh đồ họa, biểu đồ Chart cho từng nhóm sản phẩm đặt ở tiêu đề đầu trang.

Báo biểu Report mẫu này có trên CD-ROM trong thư mục Projects có tên là Top5products.rpt.

1. Mở Crystal Reports và nhấp vào mục **File > New > Standard Report**.
2. Nhấp 2 lần vào nút ODBC (RDO) để mở danh sách các nguồn dữ liệu ODBC được hỗ trợ.
3. Chọn **Xtreme Sample Database 11** và nhấp vào nút **Finish**.
4. Trong trang CSDL của Wizard, mở folder **Xtreme Sample Database** và sau đó mở mục Tables để xem các giá trị trong Tables.
5. Nhấp 2 lần vào Orders Detail và sau đó chọn tiếp bảng Products rồi di chuyển chúng qua danh sách bên phải, sau đó nhấp nút **Next**. Trang **Links** sẽ xuất hiện và mặc định rằng tất cả các liên kết bảng đã được tạo cho bạn.
6. Nhấp vào nút **Next** để tiếp tục.
7. Trong trang Fields, nhấp 2 lần vào bảng Products để mở nó và chọn trường Product Name để thêm vào danh sách các Field được chọn. Lập lại động tác này cho trường Quantity trong bảng Orders Detail và sau đó nhấp vào nút **Next**.
8. Trong trang Grouping, nhấp 2 lần vào trường Product Name để thêm chúng vào trường chọn và sau đó nhấp nút **Next**.
9. Trong trang tổng (Summary), xác định trường Quantity có xuất hiện trong danh sách chọn bên phải không, nếu không có thì nhấp 2 lần vào trường đó để đưa qua và sau đó nhấp nút **Next**.
10. Trong trang nhóm sắp xếp, chọn tùy biến sắp xếp là **Top 5** và sau đó nhấp vào nút **Next**.
11. Trong trang biểu đồ Chart, chọn tùy biến là Bar Chart và sau đó nhấp nút **Next**.
12. Do chúng ta muốn hiển thị tất cả các dữ liệu có trong bảng, trong trang lọc mẫu tin Record Selection ta không làm gì cả. Nhấp vào nút **Next** để tiếp tục.
13. Và sau cùng, trong trang khuôn mẫu chọn khuôn mẫu Corporate (Blue) và sau đó nhấp vào nút Finish để xem trước báo biểu Report của bạn.
14. Tiếp đến nhấp mục **File > Summary info** và nhập vào tiêu đề cũng như mô tả cho báo biểu Report của bạn. Nhấp vào OK, báo biểu Report của bạn sẽ giống như hình 2.21.



Hình 2.21: Báo biểu Report hoàn tất.

3. LƯU BÁO BIỂU REPORT

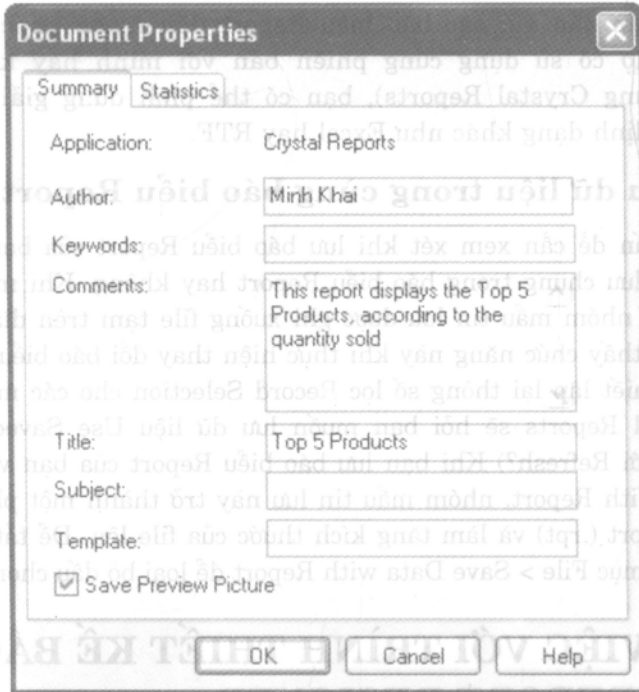
Sau khi đã hoàn tất Report Wizard, bạn trở lại Tab Preview xem trước báo biểu của mình. Tại thời điểm này, bạn sẽ quyết định xem có muốn lưu báo biểu Report trước khi chúng ta làm tiếp hay không. Bạn có thể lưu báo biểu Report của mình bằng cách nhấp vào File > Save từ menu hoặc nhấp vào biểu tượng lưu file nhanh trên thanh công cụ.

3.1. Các tùy chọn lưu file

Trên các mục chọn của menu File, mục Save Data with Report (lưu chung dữ liệu cùng với báo biểu) chọn mặc định. Chỉ có một lý do để chọn tùy biến này đó là khi bạn muốn gửi báo biểu Report của mình đến người sử dụng Crystal Reports khác mà họ không kết nối cũng như không truy xuất được CSDL của bạn. Người sử dụng khi nhận được báo biểu có thể mở và xem dữ liệu hoặc in các kết quả mà không cần đến nguồn dữ liệu nguyên thủy. Nói cách khác, mục chọn này chỉ làm cho kích thước file của Report tăng lên mà thôi.

Ngoài ra, khi chọn mục File > Summary Info (hình 2.22), bạn có thể nhập tên tác giả, từ khóa... Crystal Reports xem các trường thông tin này như những Field đặc biệt có thể dùng để thêm vào báo biểu Report của bạn. Bạn cũng có thể đánh dấu chọn ô CheckBox ở phía dưới hộp thoại để có thể tự xem trước báo biểu Report. Với chức năng này, bạn có thể xem các trang báo biểu Report dạng từng trang hay nhiều trang cùng lúc, bạn có thể duyệt trước báo biểu Report thông qua các hình ảnh thu nhỏ của trang. Thông tin này cũng được sử dụng khi bạn phát hành báo biểu Report lên Web thông qua BusinessObjects Enterprise. Các tiêu đề,

nội dung mô tả và hình ảnh thu nhỏ đều có thể xem trực tiếp từ các ứng dụng Enterprise của BusinessObjects.



Hình 2.22: Thông tin tóm lược được thêm vào báo biểu Report của bạn như một trường đặc biệt.

Ghi chú: Nếu báo biểu Report của bạn dựa vào nguồn dữ liệu luôn được cập nhật đều đặn, bạn cần làm tươi báo biểu Report của bạn mỗi khi có người muốn xem, để làm điều này hãy chọn mục Report > Refresh Report Data hoặc nhấn vào nút phím chức năng F5.

3.2. Định dạng file

Trong Crystal Reports IX, File định dạng .RPT đã hỗ trợ Unicode. Không may, định dạng file này không tương thích với các phiên bản trước của Crystal Reports. Tuy nhiên, định dạng file mới này hỗ trợ bởi Crystal Reports IX, X và XI cũng như Crystal Enterprise IX, X và BusinessObjects Enterprise XI. Vì vậy, báo biểu Report tạo trong Crystal Reports XI có thể mở sử dụng ở các phiên bản trước và ngược lại. Nhưng hãy chú ý rằng, một số chức năng sẽ không tương thích với các phiên bản trước, do nếu bạn mở báo biểu Report phiên bản XI trong Crystal

Reports X thì một vài chức năng mới trong phiên bản X sẽ không có trong phiên bản X hoặc sẽ bị loại bỏ hay xóa đi.

Nếu bạn cần gửi các báo biểu Report đến người khác mà không chắc rằng họ có sử dụng cùng phiên bản với mình hay không (hoặc không sử dụng Crystal Reports), bạn có thể phải dùng giải pháp xuất (Export) ra định dạng khác như Excel hay RTF.

3.3. Lưu dữ liệu trong cùng báo biểu Report

Một vấn đề cần xem xét khi lưu báo biểu Report của bạn là có nên đem dữ liệu lưu chung trong báo biểu Report hay không. Khi một báo biểu Report chạy, nhóm mẫu tin lưu được ghi xuống file tạm trên đĩa cứng (bạn có thể nhận thấy chức năng này khi thực hiện thay đổi báo biểu Report của mình hoặc thiết lập lại thông số lọc Record Selection cho các mẫu tin được chọn, Crystal Reports sẽ hỏi bạn muốn lưu dữ liệu Use Saved Data hay muốn làm mới Refresh?) Khi bạn lưu báo biểu Report của bạn với tùy chọn Save Data With Report, nhóm mẫu tin lưu này trở thành một phần của file báo biểu Report (.rpt) và làm tăng kích thước của file lên. Để tắt chức năng này đi, chọn mục File > Save Data with Report để loại bỏ dấu chọn mục này.

4. LÀM VIỆC VỚI TRÌNH THIẾT KẾ BÁO BIỂU REPORT DESIGNER

Với báo biểu Report đầu tiên vừa tạo, đã đến lúc chúng ta tiếp cận với môi trường thiết kế Report Designer, nơi bạn sẽ làm việc thường xuyên nhất. Trong phần này, bạn sẽ học các kỹ thuật cơ bản cần thiết trong suốt quá trình thiết kế xử lý báo biểu Report. Bạn sẽ học cách duyệt các mục báo biểu Report, chèn thêm vào (Insert), di chuyển (Move) và định dạng các đối tượng khác nhau cũng như sử dụng các kỹ thuật định dạng báo biểu thông dụng.

4.1. Môi trường Report Designer

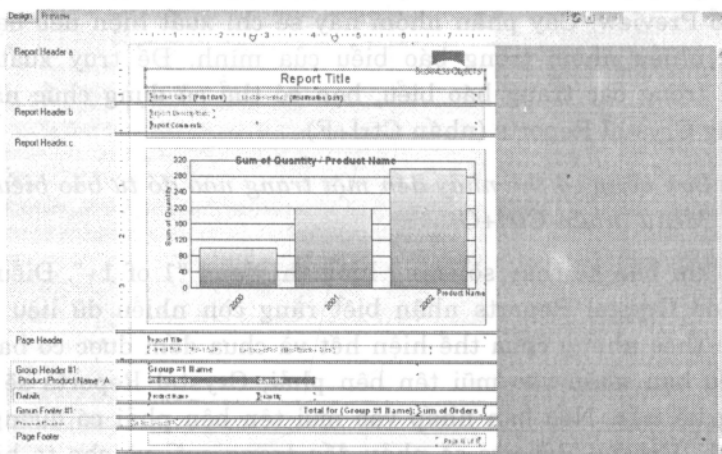
Môi trường thiết kế báo biểu Report Designer được chia thành 7 vùng, với mỗi vùng có những mục đích và những đặc tính riêng của chúng. Một số vùng trong môi trường thiết kế là chuẩn đối với tất cả các loại báo biểu như Tab Design, Tab Preview, thanh công cụ định hướng và các thanh công cụ khác.

Các vùng khác (như phần Report Section) sẽ tùy thuộc vào loại Report của bạn mà có thể xuất hiện lặp lại nhiều lần. Hiểu được những

gì sẽ diễn ra trong môi trường thiết kế báo biểu Report là chìa khóa để thiết kế báo biểu thành công. Phần này sẽ cung cấp cho bạn cái nhìn tổng quát về ý nghĩa và nội dung các vùng (section) khác nhau trong môi trường thiết kế và làm thế nào để có thể sử dụng chúng.

4.1.1. Design Tab và Preview Tab

Bạn có thể xem báo biểu Report của mình theo 2 chế độ: chế độ thiết kế (Design) và chế độ xem thử (Preview). Chế độ thiết kế (hình 2.23) cho phép xem và tương tác với những đối tượng tạo nên báo biểu Report của bạn. Nó đại diện cho một trang báo biểu tổng quát. Các thay đổi của bạn đối với các phân đoạn (Section) hoặc đối tượng đặt trong các phân đoạn khi đang thiết kế sẽ được áp cho toàn bộ báo biểu Report. Ví dụ, nếu bạn thay đổi tiêu đề trong phần đầu trang thì việc thay đổi đó sẽ xuất hiện trong tất cả phần đầu mỗi trang của báo biểu Report, bất chấp báo biểu Report của bạn có 1 hay 100 trang.



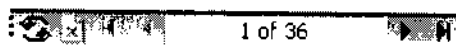
Hình 2.23: Trang thiết kế Crystal Reports.

Chế độ xem thử (Preview) cho phép bạn xem nội dung kết xuất thực thụ trước khi in theo chuẩn máy in mặc định của bạn. Nó thể hiện một cách chính xác, theo phong cách WYSIWYG (What you see is what you get = cái bạn nhìn thấy là cái bạn có được) nội dung báo biểu Report thực thụ khi thể hiện trên giấy trắng mực đen.

Thông thường thì ta làm việc nhiều trong chế độ thiết kế vì tại đây bạn có thể tác động, thay đổi, sắp xếp lại các đối tượng thành phần tạo nên nội dung báo biểu, xem được những thành phần ẩn không thể hiện trong chế độ Preview.

4.1.2. Các phương thức điều hướng

Thanh công cụ điều hướng (hình 2.24) được sử dụng để điều hướng giữa các trang trong chế độ Preview. Các ký hiệu mũi tên trong thanh công cụ sẽ di chuyển mỗi lần 1 trang báo biểu. Hình mũi tên với một đường thẳng đứng chặn lại sẽ di chuyển trang báo biểu Report của bạn về đầu hay về cuối. Hơn nữa, bạn có thể đóng cửa sổ Preview và ngừng xử lý báo biểu Report bằng cách sử dụng các nút tương ứng trong thanh công cụ điều hướng.



Hình 2.24: Thanh công cụ điều hướng dùng để di chuyển tới, lui các trang báo biểu.

Để di chuyển đến một trang báo biểu đặc biệt nào đó, bạn cũng có thể sử dụng cây phân nhóm (Group Tree), chúng được đặt ở bên cạnh trái của cửa sổ Preview. Cây phân nhóm này sẽ chỉ xuất hiện nếu bạn thêm một hoặc nhiều nhóm trong báo biểu của mình. Để truy xuất nhanh thông tin trong các trang báo biểu, bạn có thể sử dụng chức năng tìm kiếm trong Crystal Reports (nhấn Ctrl+F).

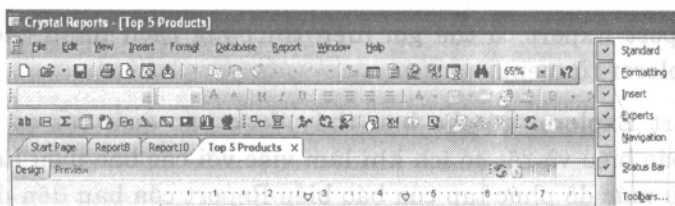
Ghi chú: Bạn cũng có thể nhảy đến một trang nào đó từ báo biểu Report Menu (nhấn Ctrl+G).

Đôi khi bạn sẽ thấy số trang hiển thị dạng "1 of 1+". Điều này có nghĩa là do Crystal Reports nhận biết rằng còn nhiều dữ liệu cho các trang tiếp theo nhưng chưa thể hiện hết và chưa đếm được có bao nhiêu trang. Nếu bạn nhấp vào mũi tên bên phải, Crystal Reports sẽ chuyển đến trang kế tiếp. Nếu bạn nhấp vào mũi tên bên phải có đường thẳng chặn lại thì Crystal Reports sẽ nhảy đến trang cuối và cho ta biết tổng số trang của báo biểu Report.

4.1.3. Thanh công cụ Toolbars

Thanh công cụ thường chứa nút cho phép truy xuất tắt (Shortcut) các mục thông dụng trên menu.

Bạn có thể thấy được bốn thanh công cụ khác nhau và thanh trạng thái bằng cách nhấp nút phải chuột trên vùng thanh công cụ và chọn loại thanh công cụ từ danh sách này (hình 2.25).



Hình 2.25: Các thanh công cụ của Crystal Reports.

- **Standard (Chuẩn):** Cung cấp các nút chuẩn của Window như Open (mở), Save (lưu), Print (in) và Refresh (làm tươi) báo biểu Report cũng như các nút Copy (sao chép), Cut (cắt), Paste (dán), Undo (hoàn tác), Redo (lập tác) và các hoạt động cơ bản của Crystal Reports. Các thanh công cụ này được hiển thị mặc định khi bạn mở Crystal Reports.
- **Formatting (Định dạng):** Cung cấp các ShortCut định dạng phổ biến, bao gồm: Font, Size (kích thước font) và Alignment (sắp thẳng hàng). Thanh công cụ này được hiển thị mặc định khi bạn mở Crystal Reports.
- **Insert (Thêm):** Bao gồm các công cụ thêm trường và các đối tượng khác như Field CSDL, các Group (nhóm) và các trường tổng kết (Summary).
- **Experts (Chuyên gia):** Cung cấp các ShortCut để sử dụng các công cụ Expert phổ biến, bao gồm Running Total Expert và Highlighting Expert.
- **Navigation (Điều hướng):** Cung cấp các điều khiển điều hướng để duyệt xem báo biểu Report bao gồm các nút di chuyển đến trang kế tiếp và trở lại trang trước đó.
- **Status Bar (Thanh trạng thái):** Hiển thị ở phía dưới trang thiết kế báo biểu Report và hiển thị tên các đối tượng, kích thước, khuôn khổ, số các mẫu tin, trạng thái, tình trạng xử lý... Thanh trạng thái được hiển thị mặc định khi bạn mở Crystal Reports.

4.2. Report Explorer

Một khái niệm khác không kém phần quan trọng để tìm hiểu trong việc tạo báo biểu Report là các báo biểu Report được ngắt thành các đoạn riêng biệt gọi là Section. Các phần riêng biệt này tạo nên báo biểu Report thể hiện ở cửa sổ trái màn hình, như trong Report Expert.

Cửa sổ Report Expert có thể gọi hiển thị bằng cách nhấp chọn View > Report Explorer từ menu.

Report Explorer sẽ cho bạn thấy được bao quát tất cả các thành phần của báo biểu và rất có ích khi làm việc với các báo biểu Report phức tạp. Cho dù mức độ phức tạp của báo biểu Report của bạn đến đâu đi nữa, các phần sau luôn thể hiện trong báo biểu Report của bạn và được sử dụng với chức năng như mô tả sau:

- **Report Header/Footer** (Phần đầu/Phần chân báo biểu): Nội dung đặt ở phần đoạn này sẽ xuất hiện ở trang đầu tiên và trang cuối cùng của báo biểu Report. Tiêu đề báo biểu Report thường xuất hiện ở trang đầu hay thường nằm ở trang cuối cùng của báo biểu Report, thêm vào đó có thể đặt ở đây phần tổng kết quan trọng cho báo biểu Report của bạn.
- **Page Header/Footer** (Phần đầu/Chân trang): Nội dung đặt ở phần đoạn này sẽ xuất hiện ở đầu và cuối mỗi trang của báo biểu. Phần đầu trang và chân trang này được dùng để hiển thị thông tin như tiêu đề các cột, số trang và ngày in báo biểu Report.
- **Group Header/Footer** (Phần đầu/Chân nhóm): Nội dung đặt ở phần đoạn này sẽ xuất hiện khi xuất hiện nhóm hoặc kết thúc nhóm. Phần đầu mỗi nhóm hay cuối nhóm thường chứa tên cột (Field), tên nhóm và cũng có thể chứa các công thức, tổng phụ của nhóm và tổng kết cơ bản của dữ liệu trong nhóm.
- **Detail** (Phần chi tiết): Đây là phần thể hiện từng dòng dữ liệu của báo biểu Report, nó trình bày nội dung của tất cả các dữ liệu trong báo biểu.

Trong Tab Design, mỗi phân đoạn Section nêu trên của báo biểu được thể hiện một dòng duy nhất, nhưng khi báo biểu Report được xem ở chế độ Preview hoặc được in ra, những phân đoạn này sẽ thực thi lặp lại dữ liệu nhiều lần để kết xuất toàn bộ nội dung trong danh sách báo cáo (hình 2.26).

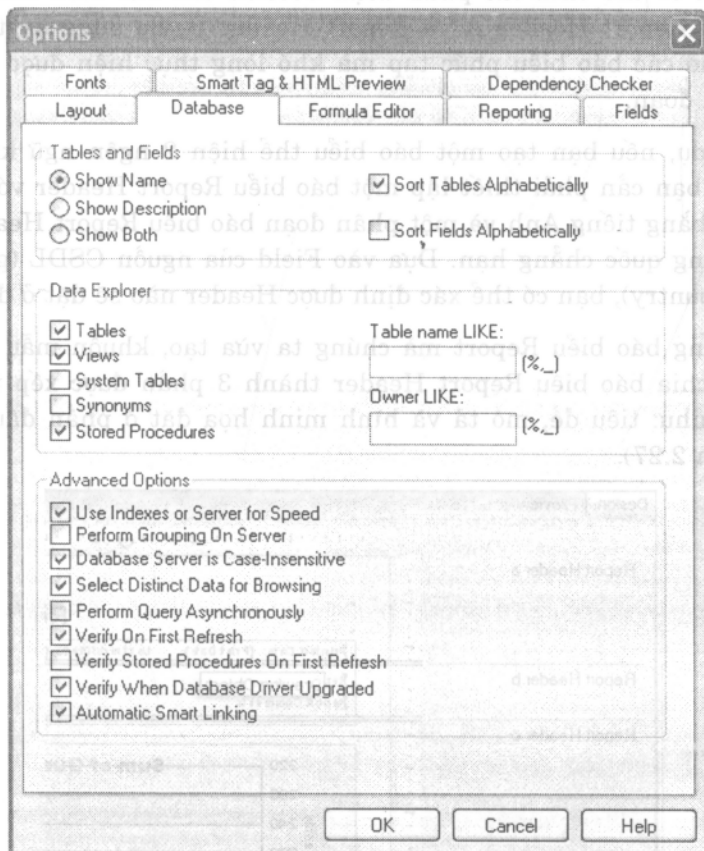
Ghi chú: Đôi khi bạn dễ nhầm lẫn khi chuyển đổi qua lại giữa chế độ thiết kế Design và xem thử Preview, đặc biệt là khi bạn thấy nội dung của báo biểu có một số Field lạ và không biết chúng xuất hiện từ đâu. Một mẹo nhỏ để giúp bạn hiểu được vấn đề là hãy kích chọn Field dữ liệu đó. Khi chuyển trở về chế độ thiết kế, bạn sẽ thấy các đối tượng đã được đánh dấu chọn và khi chuyển qua chế độ Preview bạn cũng sẽ thấy xuất hiện khung chọn đánh dấu nó.

4.3. Tùy biến môi trường thiết kế

Môi trường thiết kế có thể tùy biến giao diện cho phù hợp với công việc thiết kế báo biểu của bạn. Thiết lập của môi trường thiết kế được tổ chức thành hai nhóm tùy chọn: nhóm tùy biến toàn cục (Global Option) và nhóm tùy biến báo biểu (Report Option).

4.3.1. Global Options (Tùy biến toàn cục)

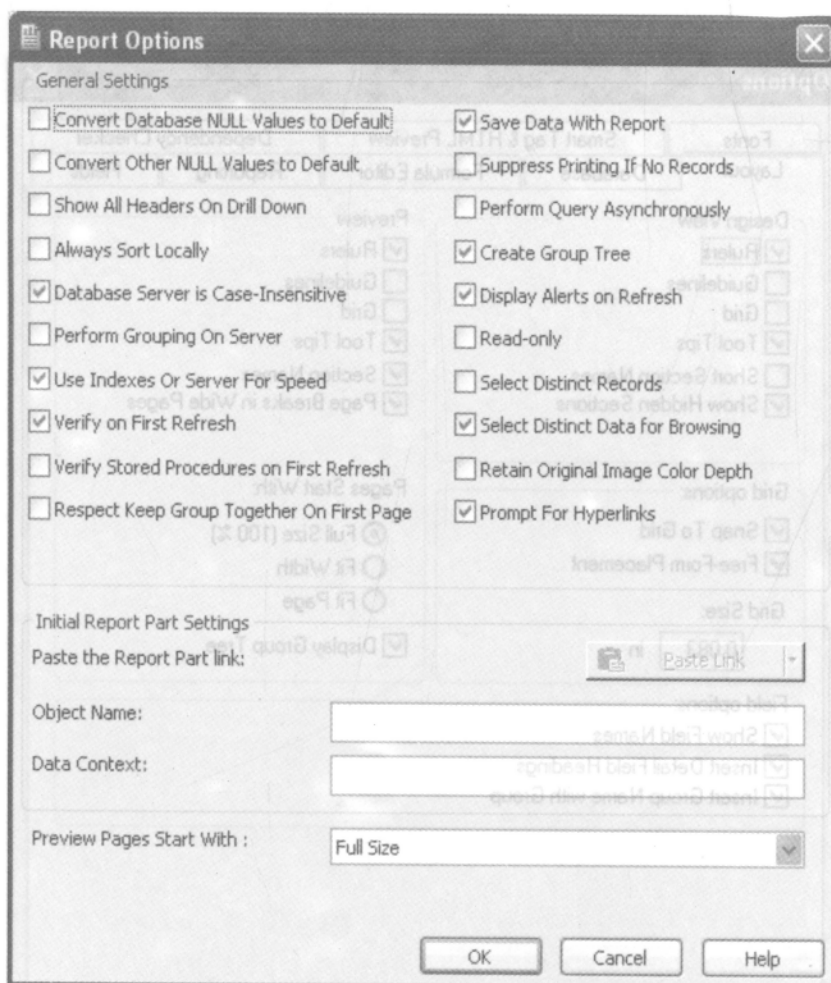
Bạn chọn mục File > Option, một hộp thoại xuất hiện cung cấp toàn bộ các tùy biến có ảnh hưởng đến tất cả các báo biểu bạn tạo ra trong Crystal Reports. Sử dụng tùy biến này bạn có thể hiệu chỉnh cách dàn trang của báo biểu Report và cửa sổ xem trước Preview, định nghĩa nơi lưu trữ báo biểu và thiết lập một số tùy biến khác áp dụng cho chế độ thiết kế báo biểu Report (hình 2.28).



Hình 2.28: Các tùy chọn toàn cục.

4.3.2. Các tùy chọn dành riêng cho Report

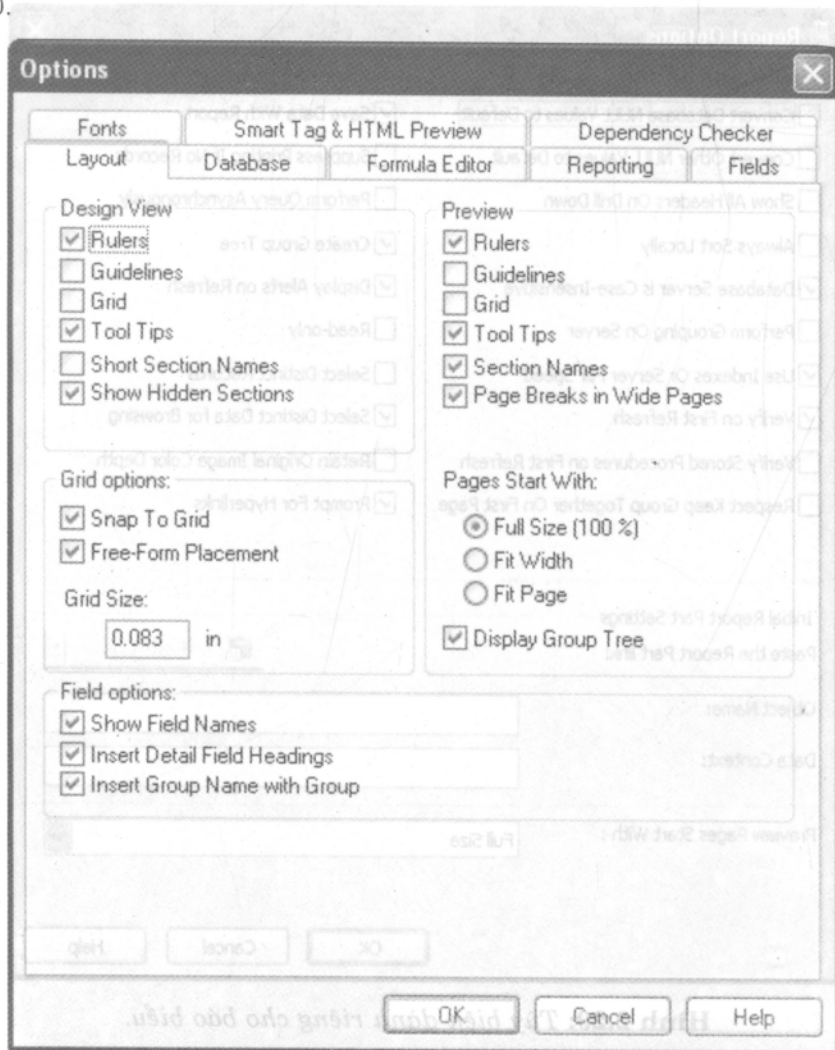
Nhấp chọn File > Report Options từ menu (hình 2.29), hộp thoại thiết lập tùy biến liên quan đến báo biểu xuất hiện. Những gì bạn thiết lập ở đây sẽ có hiệu ứng ngay lập tức. Report Options cho biết cách hiển thị các Field rỗng (null) như thế nào, có cần lưu những kết quả tính tổng chung với báo biểu hay không. Hộp thoại này cũng cho phép bạn xác định một số thuộc tính của chế độ Preview.



Hình 2.29: Tùy biến dành riêng cho báo biểu.

4.3.3. Tùy biến dàn trang (Layout)

Mỗi người có một phong cách và kiểu làm việc riêng của mình đối với một gói phần mềm nào đó. Crystal Reports đã cung cấp những đặc tính riêng đó cho bạn. Bạn có thể cấu hình môi trường thiết kế theo ý riêng của mình bằng cách nhấp chọn **File > Options > Layout** từ menu. Bạn có thể điều khiển cách hiển thị thước kẻ, phần hướng dẫn, tên các Section... Cửa sổ đặt những tùy biến dàn trang được hiển thị như hình 2.30.



Hình 2.30: Tùy biến cho môi trường thiết kế.

Với những tùy biến này bạn có thể thiết lập chúng theo ý mình. Một số người thiết kế báo biểu Report chỉ thích hiển thị những chức năng cơ bản, không cần thanh thước kẻ, các biên chỉ dẫn cũng như những thứ chiếm chỗ trên màn hình. Một số người lại cần chúng và cảm thấy dễ chịu hơn khi có các công cụ này.

Một số tùy chọn có thể tác động đến những đối tượng đang có trên báo biểu là:

- **Show Field name (Thể hiện tên Field):** Tùy biến này hiển thị tên Field đầy đủ thay vì đặt một chuỗi đại diện (Placeholder) như XXXXXXXX.
- **Insert Detail Field Headings (Thêm chi tiết Field tiêu đề):** Khi một Field được đưa vào trang báo biểu, một đối tượng Text khác sẽ được tự động thêm vào phần Header của trang để làm tiêu đề cho Field.
- **Insert Group Name with Group (Thêm vào tên nhóm):** Một tên đặt cho nhóm sẽ được chèn vào cây thư mục báo biểu giúp bạn có thể duyệt và nhảy đến từng nhóm nhanh hơn.

Bạn cũng có thể chọn tùy biến cho chế độ xem Preview như mở cửa sổ với kích thước ban đầu là toàn màn hình (FullSize 100%), theo chiều (FitWidth) rộng hay canh vừa đúng (FitPage) với kích thước trang.

Cuối cùng, để giúp cho bạn canh lề các thành phần đối tượng được đều nhau trong thiết kế báo biểu, Crystal Reports có công cụ lưới (Grid) giúp bạn có thể sắp thẳng hàng các đối tượng. Tùy biến của lưới này bao gồm các mục chọn sau đây:

- **Snap to Grid (Canh sát với lưới):** Canh lề tất cả các đối tượng được thêm vào hoặc bị xóa bỏ khiến chúng bám vào tọa độ lưới gần kề nhất.
- **Free-From Placement (Dạng sắp xếp tự do):** Cho phép bạn di chuyển và đặt để các đối tượng tự do.
- **Grid Size (Kích thước khung lưới):** Cung cấp kích thước của khung lưới dùng làm tọa độ cơ sở canh lề, các đơn vị đo bạn có thể điều chỉnh là Inches hoặc Centimeter.

5. TÓM TẮT

Crystal Reports có thể dùng để tạo báo biểu với hầu hết các loại nguồn dữ liệu khác nhau, sử dụng các kiểu kết nối trực tiếp (Native), ODBC hay thông qua trình điều khiển đặc biệt được thiết kế riêng để làm việc được chỉ với Crystal Reports. Việc kết nối một nguồn dữ liệu để xử lý là không khó khăn cho lắm nhưng hầu hết các nhà phát triển ứng dụng đều thích sử dụng trình trợ giúp Wizard để bắt đầu công việc kết nối. Trong chương kế tiếp, chúng ta sẽ học cách định dạng và cấu hình báo biểu.

Chương 3:

ĐỊNH DẠNG VÀ CẤU HÌNH BÁO BIỂU REPORT

- ❖ *Tổng quan.*
- ❖ *Tìm hiểu các đối tượng Field.*
- ❖ *Chèn vào các đối tượng Field.*
- ❖ *Cách hoạt động của các đối tượng Field.*
- ❖ *Cách hoạt động của các phân đoạn (Section).*
- ❖ *Cách định dạng các phân đoạn (Section).*
- ❖ *Thêm các nhóm (Group) vào báo biểu Report.*
- ❖ *Tùy biến định dạng nhóm.*
- ❖ *Thay đổi các nhóm.*
- ❖ *Nhóm theo chỉ định.*
- ❖ *Nhóm các Field dạng ngày giờ.*
- ❖ *Sắp xếp các mẫu tin theo cấp.*
- ❖ *Đào sâu (Drill-Down) và báo cáo dạng tổng.*
- ❖ *Tóm tắt.*

1. TỔNG QUAN

Định dạng là một phần quan trọng của xử lý thiết kế báo biểu. Các nhà phát triển báo biểu phải mất rất nhiều thời gian để định dạng báo biểu, điều chỉnh giao diện của nó cho phù hợp những gì họ cần. Trong chương này chúng ta sẽ học các kỹ thuật định dạng mà bạn có thể sử dụng để hiệu chỉnh báo biểu, làm việc với Field (trường) và phân đoạn (Section).

Ngoài việc định dạng, chúng ta cũng sẽ học cách thêm một giá trị dữ liệu vào trong báo biểu Report của bạn. Một trong những mục đích chính của thiết kế báo biểu Report là trích rút và tổ chức dữ liệu. Thay vì chỉ liệt kê dữ liệu trong một bảng CSDL, bạn có thể sử dụng Crystal

Reports biến dữ liệu thành thông tin mà mọi người có thể sử dụng để ra quyết định về một vấn đề nào đó được tốt hơn. Nhóm và sắp xếp dữ liệu đóng vai trò quan trọng nhất trong tổ chức báo biểu Report của bạn và tăng thêm giá trị của thông tin trình bày.

Trong phần cuối của chương này, bạn sẽ học cách tổ chức nhóm và sắp xếp báo biểu Report như phân cấp và xác định nhóm. Bạn cũng sẽ học cách áp dụng các kỹ thuật để tạo báo biểu được tổ chức tốt hơn từ nguồn dữ liệu của mình, chẳng hạn kỹ thuật Drill-Down (đào sâu) và tổng kết báo biểu Report.

2. TÌM HIỂU CÁC ĐỐI TƯỢNG FIELD

Trước khi nghiên cứu chi tiết vào phần định dạng (Formatting) báo biểu, chúng ta cần xem qua các thành phần cơ bản tạo nên một báo biểu, đó là các đối tượng Field (còn gọi là Field Object). Một báo biểu có thể chứa nhiều đối tượng Field khác nhau. Field có thể xuất phát từ các trường của bảng trong CSDL, Field có thể là biến, Field cũng có thể nắm giữ văn bản (Text) mà bạn nhập trực tiếp vào từ báo biểu. Bạn có thể phải sử dụng tất cả các kiểu đối tượng Field này để xây nên báo biểu mà mình mong muốn. Phần này sẽ mô tả các đối tượng Field thông dụng, cách bạn nhận biết và sử dụng chúng trong báo biểu Report

2.1. Field CSDL (Database Field)

Các Field CSDL được lấy ra từ bảng dữ liệu, View hay Store Procedure được sử dụng trong báo biểu. Các Field CSDL được thể hiện ký hiệu trong chế độ thiết kế là {TableName.Field}. Thông tin này có thể được nhìn thấy bằng cách nhấp vào Field và nhìn vào góc trái bên dưới thanh trạng thái. Ví dụ, nếu bạn thấy Field **Phone Number** từ bảng **Customer**, ký hiệu của nó sẽ là {Customer.Phone}. Các Field CSDL được dùng để hiển thị thông tin từ bảng CSDL của bạn và được dùng thông dụng trong các phần đoạn chi tiết (Detail Section) của báo biểu, mặc dù vậy chúng cũng có thể xuất hiện ở một nơi nào đó khác trong báo biểu.

2.2. Field tiêu đề (Heading Field)

Khi bạn thêm một Field CSDL vào phần đoạn chi tiết (Detail Section) của báo biểu, một đối tượng nhãn được thêm vào để làm tiêu đề mô tả cho Field CSDL. Đối tượng nhãn này gọi là Field tiêu đề. Khi bạn di chuyển Field CSDL, Field tiêu đề sẽ di chuyển theo.

2.3. Field công thức (Formula Field)

Field công thức là một Field tính toán có thể thêm vào báo biểu và được hiển thị như mọi Field. Kết hợp sức mạnh của ngôn ngữ lập trình như Pascal, Excel và Visual Basic, công thức có thể sử dụng chung với các Field CSDL, các biến tham số ... để thực thi các tính toán phức tạp dạng số, chuỗi, ngày và thời gian.

Field công thức luôn được đặt một biểu tượng @ vào trước tên Field và được bao bởi dấu ngoặc móc. Ví dụ, {@commission}. Field công thức (có thể được thêm vào nhiều nơi trong báo biểu Report) được sử dụng rộng rãi trong tính toán số học, thao tác chuỗi và thực thi câu lệnh logic phức tạp để cho ra kết quả tổ hợp.

2.4. Field tham số biến (Parameter Field)

Các Field tham số được dùng để yêu cầu người dùng nhập thông tin cho báo biểu. Các Field tham số được thêm dấu "?" trước tên Field và được bao bởi dấu ngoặc móc. Ví dụ, {?EnterState}. Các Field tham số có thể được sử dụng với cơ chế lọc mẫu tin Record Selection, kết hợp trong công thức... và có thể được thêm vào bất cứ chỗ nào trong báo biểu Report.

2.5. Field đặc biệt (Special Field)

Các Field đặc biệt được Crystal Reports sinh ra như Field dùng đánh số trang, các Field tính thông tin tổng gộp. Các Field đặc biệt được ký hiệu chỉ bằng tên của chúng và tất cả các tên Field được quy định dành riêng chỉ có hệ thống được phép sử dụng. Các Field đặc biệt có thể thêm vào bất cứ nơi nào trong báo biểu Report.

2.6. Các Field tổng chạy dồn (Running Total Field)

Field tổng "chạy dồn" là một Field tính tổng chuyên dụng có thể sử dụng để tạo các số tính dồn theo tổng, trung bình cộng, v.v... và hiển thị thông tin này trên báo biểu Report. Field tổng chạy dồn này được thêm biểu tượng # trước tên Field và được bao bởi dấu ngoặc nhọn, ví dụ như {#TotalSales}. Tổng chạy dồn thường hiển thị trong phần chân trang (cuối một trang) hay trong phân đoạn chi tiết dữ liệu (cuối dòng phân đoạn chi tiết dữ liệu).

2.7. Các Field biểu thức SQL (SQL Expression Field)

Field biểu thức SQL giống như là Field công thức của Crystal Reports, các Field biểu thức SQL có thể sử dụng cho tính toán. Tuy nhiên, với Field biểu thức SQL, các tính toán xảy ra trên CSDL Server tận dụng được lợi thế sức mạnh xử lý của Server. Một Field biểu thức SQL được ký hiệu bằng dấu % trước tên Field và được bao bởi dấu ngoặc móc, ví dụ như {**%CalcSummary**}. Field dạng biểu thức SQL có thể thêm vào bất cứ nơi nào trong báo biểu Report.

2.8. Các Field tổng (Summary Field)

Field tổng có thể được sử dụng cho tính toán đơn giản như tổng con (SubTotal) hoặc tính trung bình cộng, độ lệch tiêu chuẩn, mật độ phân bố... Thoạt nhìn, các Field tổng và Field công thức cùng làm một việc tính toán, nhưng khác nhau chủ yếu là các Field tổng cần yêu cầu viết mã. Field tổng có thể nhận dạng bằng cách sử dụng các kiểu công thức tổng (Sum, Average v.v...) với từ khóa **OF** cùng với tên Field được đem ra tính tổng, ví dụ: Sum of Sales. Field tổng thường được đặt cuối nhóm (Group), cuối trang (Page) hoặc tại tiêu đề của báo biểu (Report Header) hay tại phần chân (Footer).

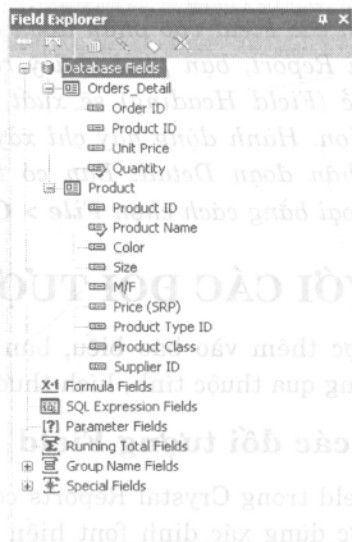
2.9. Field tên nhóm (Group Name Field)

Field tên nhóm được sinh ra bởi Crystal Reports để tạo nhãn cho các nhóm mà bạn thêm vào báo biểu Report. Tất cả các Field tên nhóm được đặt theo khuôn dạng – Group #n Name – giá trị n là số của nhóm mà bạn làm việc, ví dụ như **Group #3**. Các tên nhóm được tự động thêm vào tương thích với tiêu đề nhóm (Group Header) hay phần chân nhóm (Group Footer). Khi một Field tên nhóm hiển thị trên Tab Preview của báo biểu Report, nó sẽ mang tên của nhóm mà bạn đã thêm vào.

3. CHÈN ĐỔI TƯỢNG FIELD VÀO BÁO BIỂU

Một trong những công việc phổ biến khi tạo báo biểu là chèn thêm các Field vào báo biểu. Để thêm các Field vào báo biểu bạn sử dụng cửa sổ **Field Explorer** bằng cách nhấp chọn từ menu **View > Field Explorer**. Cửa sổ Field Explorer được hiển thị như hình 3.1.

***Ghi chú:** Bạn có thể thay đổi kích thước Field Explorer bằng cách kéo co giãn góc phải bên dưới của cửa sổ Field Explorer.*



Hình 3.1: Crystal Reports Field Explorer.

Field Explorer hiển thị tất cả các loại Field mà bạn có thể thêm vào báo biểu. Các Field được thể hiện theo thứ bậc từ trên xuống. Bạn có thể thấy các Field của mỗi loại bằng cách nhấp vào ký hiệu dấu cộng kế bên tên phân loại của Field để mở rộng nhóm phân loại đó ra.

Nếu bạn nhấp vào ký hiệu dấu cộng kế bên kiểu Field CSDL (DataBase Fields), một danh sách các bảng có trong CSDL mà bạn chọn cho báo biểu sẽ được hiển thị. Để tìm Field bạn cần, bạn phải mở rộng nội dung mỗi bảng dữ liệu. Các loại Field khác bạn cũng có thể thực hiện tương tự.

Ghi chú: Nếu bạn có số lượng lớn các bảng dữ liệu và Field, bạn cần phải nghĩ ngay đến việc sắp xếp theo thứ tự ABC tên các bảng dữ liệu, tên các Field hoặc cả hai. Bạn có thể tìm thấy tùy biến này bằng cách nhấp vào mục **File > Options > Database** từ menu.

Bạn sẽ mất rất nhiều thời gian, công sức cho công việc thêm vào và sắp xếp trật tự các Field trong báo biểu Report. Crystal Reports đã cố gắng làm cho công việc này dễ dàng hơn với các công cụ xử lý trực quan. Các Field có thể được kéo trực tiếp từ Field Explorer vào báo biểu Report hoặc bạn có thể đánh dấu một Field và nhấn phím Enter để gắn Field vào đầu trỏ chuột trong báo biểu Report. Sau đó, bạn có thể bố trí lại nơi mà bạn muốn đặt Field bằng cách nhấp giữ chuột và di chuyển đến chỗ cần đặt rồi thả ra.

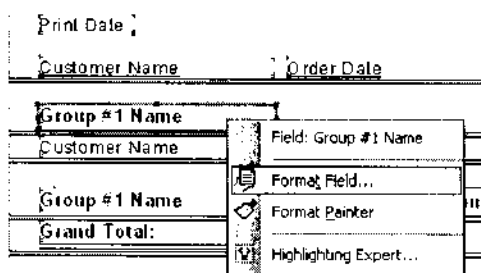
Ghi chú: Khi bạn thêm các Field vào phân đoạn chi tiết (Detail Section) của báo biểu Report, bạn phải chú ý rằng có thêm đối tượng Field tiêu đề (Field Heading) sẽ xuất hiện trong phân đoạn Header Section. Hành động này chỉ xảy ra với các Field thêm vào trong phân đoạn Detail. Bạn có thể tắt chức năng này trong hộp thoại bằng cách chọn **File > Options**.

4. LÀM VIỆC VỚI CÁC ĐỐI TƯỢNG FIELD

Khi Field đã được thêm vào báo biểu, bạn có thể điều khiển các cách hiển thị Field thông qua thuộc tính, kích thước, kiểu chữ ...

4.1. Định dạng các đối tượng Field

Mỗi đối tượng Field trong Crystal Reports có khá nhiều thuộc tính, từ thuộc tính Font được dùng xác định font hiển thị nội dung của Field cho đến các thuộc tính dùng để định dạng số, ngày tháng. Để xem thuộc tính của Field bạn nhấp chuột phải vào Field và chọn **Format Field** từ menu tắt xuất hiện (hình 3.2).



Hình 3.2: Bạn có thể định dạng Field bằng cách nhấp nút phải chuột vào Field và chọn **Format Field**.

Cửa sổ thuộc tính của đối tượng này xuất hiện. Trong ví dụ này, chúng ta chọn Field kiểu chuỗi (String Field), bạn sẽ thấy trong cửa sổ này xuất hiện các Tab Common (các thuộc tính thông dụng), Border (vẽ khung), Font (kiểu chữ) và thuộc tính Hyperlink (tạo liên kết Web). Những Tab thuộc tính sẽ xuất hiện phụ thuộc vào kiểu Field mà bạn chọn. Ví dụ, các đối tượng Field trong Crystal Reports đều hiển thị những Tab thuộc tính chung như Common, Border, Font và Hyperlink.

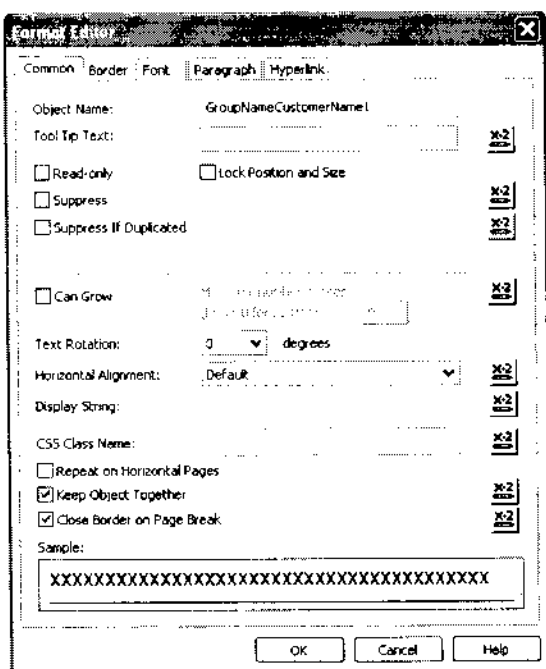
Tuy nhiên, một Field kiểu số với một Field kiểu ngày có thêm những thuộc tính khác nhau. Đối với Field kiểu số, Tab thuộc tính liên

quan đến định dạng số sẽ được thêm vào. Field kiểu ngày sẽ có thêm các Tab định dạng thời gian...

Phần dưới đây sẽ mô tả một số thuộc tính định dạng thông dụng nhất của các Field trong báo biểu.

4.1.1. Các tùy chọn định dạng thông dụng

Tất cả các loại Field trong báo biểu đều có những thuộc tính chung thể hiện trong Tab Common của cửa sổ thuộc tính (hình 3.3). Những thuộc tính này bao gồm:



Hình 3.3: Các thuộc tính định dạng phổ biến.

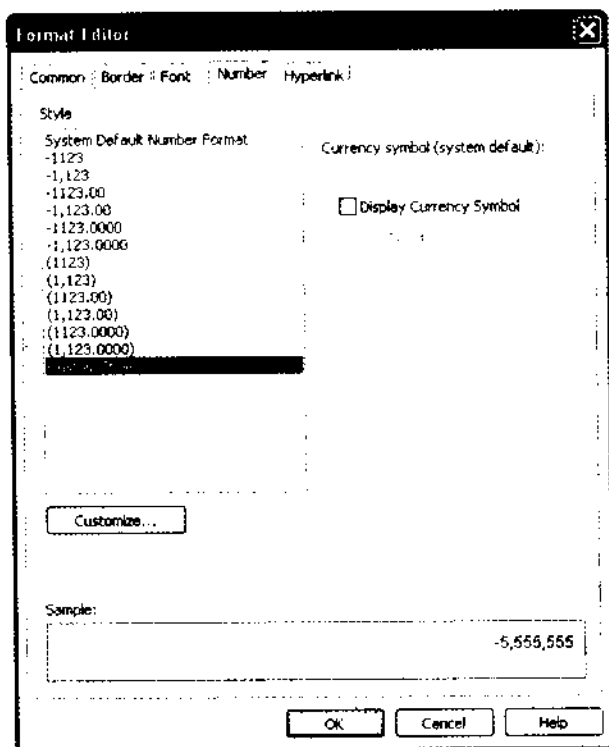
- **Object Name:** Là tên của đối tượng. Crystal Reports sẽ tự tạo tên cho đối tượng Field, sau đó bạn có thể thay đổi tên này trong báo biểu theo cách của mình sao cho có ý nghĩa hơn.
- **Tool Tip Text:** Chú giải ngắn xuất hiện khi người sử dụng di chuyển con trỏ chuột lên đối tượng. Bạn có thể nhập văn bản trực tiếp vào ô TextBox hoặc kích nút **X+2** ở vị trí phía phải. Bạn cần phải nhập chú giải vào công cụ chú giải trong cặp dấu nháy (ví dụ: "Đây là lời chú giải").

- **Read-only (Chỉ đọc):** Thiết lập này xác định thuộc tính đối tượng có được thay đổi hay không.
- **Lock Position and Size (Khóa vị trí và kích thước):** Thiết lập này khóa đối tượng dính chặt vào một vị trí cố định nào đó trên báo biểu không cho thay đổi kích thước.
- **Suppress (Ngăn hiển thị):** Thiết lập này sẽ khiến cho đối tượng không thể xuất hiện trong báo biểu Report (đối tượng sẽ vẫn còn trên báo biểu Report trong chế độ Design nhưng sẽ không hiển thị khi xem ở chế độ Preview hoặc in ra giấy).
- **Suppress If Duplicated (Ngăn hiển thị bị trùng lặp):** Sử dụng thiết lập để ngăn Field hiển thị nếu nội dung bị trùng lặp (2 mẫu tin giống hệt nhau).
- **Can Grow (Có thể giãn ra):** Đối với các đối tượng chứa nội dung văn bản nhiều dòng, chọn thiết lập này để đảm bảo rằng đối tượng có thể giãn ra khi cần để chứa đủ nội dung gồm 2 dòng hay 20 dòng. Để điều khiển được kích thước của đối tượng, bạn có thể thiết lập một số dòng cực lớn. Mặc định rằng, chỉ số này được thiết lập bằng 0 cho biết là không giới hạn.
- **Text Rotation (Quay văn bản):** Sử dụng thiết lập này để quay đối tượng văn bản 90 độ hay 270 độ.
- **Horizontal Alignment (Canh lề ngang):** Chọn từ danh sách Drop-Down các giá trị canh trái (Left-align), canh giữa (Center), canh phải (Right-Align) hoặc canh đều (Justify) nội dung đối tượng. Mặc định của thiết lập này sẽ tùy theo từng kiểu của đối tượng.
- **CSS Class Name:** Nếu bạn từng sử dụng báo biểu Report này trên trang Web, một file .CSS (Cascading Style Sheet) có thể được kết hợp với báo biểu Report. CSS thường được áp dụng cho các định dạng nhất quán xuyên suốt các trang Web.
- **Repeat on Horizontal Pages (Lập lại nội dung kết xuất theo chiều ngang):** Tùy biến này được thiết lập để lập lại đối tượng trên các trang theo chiều ngang.
- **Keep Object Together (Giữ đối tượng đồng nhất):** Tùy biến này được thiết lập để cố gắng giữ cho các đối tượng lớn cùng nằm trên một trang.

- **Close Border on page Break (Đóng đường viền trên trang ngắt):** Thiết lập tùy biến này cho phép các đối tượng có khung viền quanh được xem là đóng khung ở cuối trang trước khi sang trang kế tiếp.

4.1.2. Định dạng số và tiền tệ

Để định dạng số và các loại tiền tệ, Crystal Reports cung cấp một số thuộc tính riêng. Bạn có thể thiết lập các thuộc tính định dạng số theo những mẫu có sẵn (hình 3.4).



Hình 3.4: Chọn định dạng số theo mẫu.

Để thêm biểu tượng tiền tệ, sử dụng tùy biến ở góc trên bên phải của hộp thoại. Bạn cũng có thể chỉ định biểu tượng sẽ bố trí ở vị trí nào đó hoặc thay đổi bên cạnh các số.

Để chỉ định một định dạng số tùy biến, nhấp vào nút **Customize** ở phía dưới hộp thoại. Lệnh này cho phép tùy biến các giá trị biểu tượng tiền tệ và số như sau:

- **Enable Currency Symbol (Cho phép biểu tượng tiền tệ hoạt động):** Chỉ định cách mà biểu tượng tiền tệ sẽ hiển thị.
- **Fixed/Floating (Cố định/di động):** Chỉ định cách mà các biểu tượng được cố định một chỗ ở lề trái của Field hoặc di động sát đến con số đầu tiên.
- **One Symbol per Page (Một biểu tượng mỗi trang):** Đặt biểu tượng tiền tệ chỉ ở đầu mỗi trang.
- **Currency Symbol (Biểu tượng tiền tệ):** Chỉ định biểu tượng tiền tệ sẽ được sử dụng với các Field số.
- **Use Accounting Format (Sử dụng định dạng theo kế toán):** Đặt biểu tượng tiền tệ bên trái số và hiển thị số âm có dấu gạch ngang.
- **Suppress If Zero (Ngắt nếu bằng 0):** Các Field hiển thị sẽ bị ngắt nếu giá trị các Field đó bằng 0.
- **Decimals (Thập phân):** Chỉ định độ chính xác của số thập phân.
- **Rounding (Làm tròn):** Chỉ định số chữ số làm tròn, từ 10 đến 1.000.000 chữ số thập phân.
- **Negative (Dấu âm):** Hiển thị biểu tượng dấu âm (-) trước hoặc sau các số sử dụng dấu ngoặc đơn.
- **Reverse Sign for Display (Đảo dấu hiển thị):** Đảo dấu ký hiệu âm sẽ xuất hiện (tức là hiển thị dấu âm cho số dương và ngược lại).
- **Allow Field Clipping (Cho phép xén nội dung Field):** Chỉ định cách thực hiện cắt xén nội dung Field nếu nó lớn hơn không gian hiển thị. Crystal Reports xén Field theo mặc định và chỉ hiển thị một phần của Field. Nếu bạn bỏ chọn tùy biến này, Crystal Reports hiển thị ký hiệu (####) để chỉ ra rằng Field đó là quá lớn so với khoảng trống định dùng chứa nó. Để làm mất đi các ký hiệu đó, bạn cần kéo cấu trúc Field sao cho nó đủ lớn để đủ chỗ cho nội dung nhập vào.
- **Decimal Separator (Dấu phân cách thập phân):** Để thay đổi dấu phân cách thập phân. Mặc định rằng, Crystal Reports sử dụng dấu chấm để đánh dấu phân cách giữa số nguyên và số thập phân.

- **Thousands Separator (Dấu phân cách hàng ngàn):** Cho phép thay đổi dấu phân cách hàng ngàn và biểu tượng dấu phân cách. Mặc định rằng, Crystal Reports sử dụng dấu phẩy làm dấu phân cách hàng ngàn trong dãy số.
- **Leading Zero (Số 0 dẫn đầu):** Thêm các chữ số 0 vào các chữ số thập phân (số thập phân > 0).
- **Show Zero values As ():** Hiển thị các giá trị 0 trong mặc định hoặc thay bằng dấu (-).

Ghi chú: Có một số cách định dạng tùy biến mà bạn chỉ định không thể lưu được và phải tạo lại mỗi khi muốn sử dụng chúng.

4.1.3. Các Field định dạng ngày

Crystal Reports cung cấp cho bạn các định nghĩa định dạng ngày, tháng. Nếu bạn muốn định dạng ngày và thời gian theo vị trí địa lý của bạn, nhấp chuột phải lên Field và chọn Field định dạng, chú ý rằng thuộc tính kèm theo Tab **Date/Time** chỉ cho phép bạn chọn kiểu ngày định dạng theo mẫu có sẵn (tương tự như bạn đã làm với định dạng số vậy).

Để chọn một kiểu tùy biến cho định dạng ngày, bạn nhấp vào nút **Customize** ở phía dưới hộp thoại và có thể chọn các kiểu định dạng sau:

- *Date/Time.*
- *Date type.*
- *Calendar type.*
- *Format (Month, Day, Year).*
- *Era/Period Type.*
- *Order.*
- *Day of week type.*
- *12/24 hour.*
- *AM/PM breakdown.*
- *Symbols position.*
- *Format (Hour, Minute, Second).*
- *Separators.*

4.1.4. Kết hợp lại

Với những tùy biến định dạng trên đây, bạn có thể làm cho giao diện báo biểu trông chuyên nghiệp hơn. Trong phần thực tập sau, chúng ta sẽ định dạng các đối tượng Field của một bản báo cáo tổng kết kiểm kê kho có sẵn (hình 3.5) như hủy bỏ dấu phân cách hàng ngàn trong Field **Purchase Order**, định dạng Field ngày **Print Date** và **Order Date** đặt ở đầu báo biểu Report. Hãy làm theo các bước sau:

Inventory Summary Report				
5/25/2005				
Product Name	Reorder Level	Units in Stock	Units on Order	P.O.# Order Date
Active Outdoors Crochel Glove	300	220	500	1,001 4/1/2004 12:00:00AM
Active Outdoors Crochel Glove	300	150	0	
Active Outdoors Crochel Glove	300	325	0	
Active Outdoors Crochel Glove	300	265	0	
Active Outdoors Crochel Glove	300	367	0	
Active Outdoors Lycra Glove	300	440	0	1,002 4/3/2004 12:00:00AM
Active Outdoors Lycra Glove	300	358	0	
Active Outdoors Lycra Glove	300	266	500	
Active Outdoors Lycra Glove	300	750	0	
Active Outdoors Lycra Glove	300	112	500	
Active Outdoors Lycra Glove	300	886	0	1,003 4/5/2004 12:00:00AM
Triumph Pro Helmet	100	78	150	
Triumph Pro Helmet	100	80	0	
Triumph Pro Helmet	100	55	150	
Triumph Pro Helmet	100	159	0	
Triumph Pro Helmet	100	200	0	1,004 4/8/2004 12:00:00AM
Triumph Pro Helmet	100	168	0	
Triumph Vertigo Helmet	100	89	0	
Triumph Vertigo Helmet	100	125	0	
Triumph Vertigo Helmet	100	220	0	
Triumph Vertigo Helmet	100	150	0	1,004 4/8/2004 12:00:00AM
Triumph Vertigo Helmet	100	92	150	
Triumph Vertigo Helmet	100	127	0	

Hình 3.5: Một báo biểu tổng kết kiểm kê kho.

1. Mở Crystal Reports và mở file báo biểu Report **Inventorysummary.RPT** trên đĩa CD-ROM.
2. Nhấp vào **Design Tab** để chuyển qua chế độ thiết kế báo biểu Report.
3. Tìm Field **PO#** trong phân đoạn chi tiết (Detail Section) của báo biểu, nhấp phải chuột vào Field đó và chọn định dạng **Format Field**.
4. Nhấp vào nút **Customize**, hủy chọn tùy biến Thousands Separator, sau đó nhấp 2 lần nút **OK**.
5. Bước kế tiếp, nhấp nút phải chuột vào Field **Order Date** trong phân đoạn chi tiết và chọn định dạng **Format Field**.
6. Chọn tùy biến dạng 03/09/99 và nhấp vào **OK**.
7. Kế đến tìm Field **Print Date** trong phân đoạn tiêu đề (Page Header), nhấp phải chuột và chọn **Format Field**.

8. Chọn tùy biến dạng **Monday, March 1 1999** trong danh sách và sau đó nhấp vào nút **OK**.
9. Sau cùng nhấp vào Tab **Preview** của báo biểu, bạn sẽ thấy báo biểu Report thay đổi như hình 3.6.

Inventory Summary Report

Tuesday, 24 May 2005

Product Name	Reorder Level	Unit in Stock	Units on Order	EQ#	Order Date
Active Outdoors Crochet Glove	300	220	590	1001	04/04/2004
Active Outdoors Crochet Glove	300	450	0		
Active Outdoors Crochet Glove	300	325	0		
Active Outdoors Crochet Glove	300	255	0		
Active Outdoors Crochet Glove	300	367	0		
Active Outdoors Lycra Glove	300	440	0		
Active Outdoors Lycra Glove	300	350	0		
Active Outdoors Lycra Glove	300	265	590	1002	04/03/2004
Active Outdoors Lycra Glove	300	750	0		
Active Outdoors Lycra Glove	300	112	500	1002	04/03/2004
Active Outdoors Lycra Glove	300	566	0		
Triumph Pro Helmet	100	78	150	1003	04/05/2004
Triumph Pro Helmet	100	80	0		
Triumph Pro Helmet	100	55	150	1003	04/05/2004
Triumph Pro Helmet	100	169	0		
Triumph Pro Helmet	100	200	0		
Triumph Pro Helmet	100	168	0		
Triumph Vertigo Helmet	100	89	0		
Triumph Vertigo Helmet	100	125	0		
Triumph Vertigo Helmet	100	220	0		
Triumph Vertigo Helmet	100	150	0		
Triumph Vertigo Helmet	100	92	150	1004	04/08/2004
Triumph Vertigo Helmet	100	107	0		

Hình 3.6: Hoàn thành báo biểu Report với việc áp dụng định dạng.

4.2. Điều chỉnh kích thước các Field

Tất cả các Field bạn thêm vào trong báo biểu Report đều có thể thay đổi kích thước. Nếu nhấp vào một đối tượng Field, bạn sẽ thấy 4 khung viền điều khiển xuất hiện bao mỗi cạnh của đối tượng. Bằng cách di chuyển những đường viền điều khiển này bạn có thể thay đổi kích thước đối tượng.

Ghi chú: Việc thay đổi kích thước các đối tượng trong Crystal Reports được thao tác tương tự như thay đổi kích thước các đối tượng trong phần mềm ứng dụng Microsoft Office.

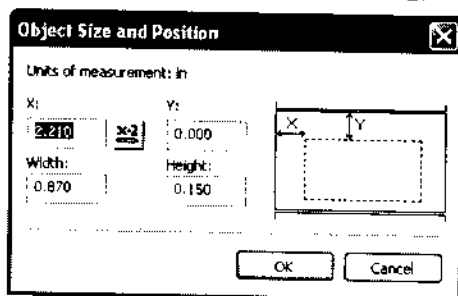
Một chức năng giúp bạn đỡ tốn thời gian đó là bạn có thể chọn nhiều đối tượng (các kiểu Field khác nhau) và khi bạn thay đổi kích thước một đối tượng nào đó, tất cả các đối tượng được chọn sẽ thay đổi theo tương ứng. Để thay đổi kích thước cho nhiều đối tượng cùng lúc, bạn sử dụng các bước sau đây:

1. Tìm đối tượng trên báo biểu Report mà bạn muốn thay đổi kích thước, chọn các đối tượng cùng lúc bằng cách kéo một khung bao quanh các đối tượng hoặc bằng cách nhấp chọn từng đối tượng trong khi giữ phím **Shift** hoặc **Ctrl**.

Ghi chú: Bạn cũng có thể chọn tất cả các đối tượng có trong phân đoạn liên quan bằng cách chuyển đến Design Tab và nhấp phải chuột lên vùng màu xám phía bên cạnh trái của màn hình phân đoạn bạn muốn chọn. Từ menu ShortCut bạn chọn mục *Select All Objects*.

2. Chọn một đối tượng cụ thể nào đó trong số các đối tượng đang được đánh dấu chọn. Kéo kích thước đối tượng đó co giãn, tất cả những đối tượng còn lại đều thay đổi theo với tỉ lệ tương ứng.
3. Nhấp bất kỳ chỗ nào ngoài các Field được chọn để hoàn tất quá trình.

Nếu cần thay đổi kích thước Field với độ chính xác nào đó, bạn cũng có thể định rõ kích thước và vị trí của nó. Tại vị trí mà bạn muốn thay đổi kích thước hoặc vị trí, nhấp phải chuột lên nó. Từ ShortCut Menu xuất hiện, chọn **Size and Position**. Sử dụng hộp thoại xuất hiện như hình 3.7 chọn vị trí **X** và **Y** của đối tượng cũng như chiều rộng, chiều cao của nó.



Hình 3.7: Bạn có thể định rõ kích thước và vị trí của đối tượng.

4.3. Di chuyển các đối tượng Field

Có nhiều cách di chuyển đối tượng Field. Trước tiên là kỹ thuật kéo thả. Bằng cách nhấp chọn vào một đối tượng, giữ nút chuột và di chuyển chuột, bạn có thể kéo các đối tượng Field đi và thả chúng ở chỗ mà bạn thích. Một phương thức khác đó là sử dụng các đường định hướng GuideLine.

Ghi chú: Một cách khác cũng có thể di chuyển đối tượng Field trong thiết kế báo biểu Report là nhấp vào đối tượng chọn chúng sau đó nhích đối tượng đi bằng cách sử dụng các phím mũi tên.

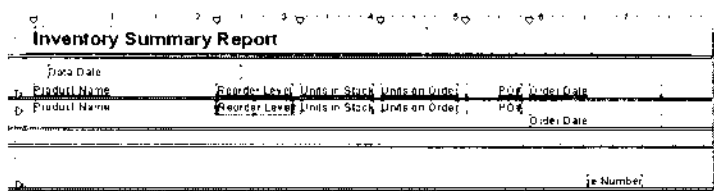
4.3.1. GuideLine (Đường định hướng)

GuideLine (hình 3.8) là đối tượng không thể trông thấy mà bạn có thể sử dụng để canh lề và di chuyển các Field. Cách dễ dàng nhất để

hình dung GuideLine là một chuỗi; bạn gắn các đối tượng vào chuỗi và khi chuỗi di chuyển, mọi thứ gắn trong chuỗi sẽ di chuyển theo.

Ghi chú: GuideLine không thấy được khi bạn in Report. Khi làm việc trong báo biểu Report thiết kế hoặc xem trước, GuideLine có thể xuất hiện 1 đường gạch thẳng đứt phân đoạn trong báo biểu Report.

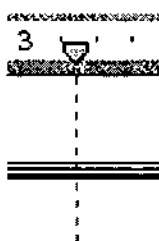
GuideLine có thể được thêm vào báo biểu Report bằng cách nhấp vào bất cứ nơi nào trên thanh thước đo. Một biểu tượng nhỏ (đôi khi gọi là Caret) sẽ xuất hiện như hình 3.9, cho biết rằng bạn đã tạo một GuideLine.



Hình 3.8: GuideLine có thể xuất hiện ở cả hai Tab thiết kế và xem trước.

Ghi chú: GuideLine cũng được tạo tự động khi bạn thêm Field mới vào phân đoạn chi tiết của báo biểu Report.

Sau khi tạo xong GuideLine, bạn có thể canh đối tượng bám sát vào GuideLine như thể có lực hút tác động vào đối tượng khiến chúng dính vào GuideLine khi bạn di chuyển.



Hình 3.9: GuideLine được tạo bằng một dấu Caret.

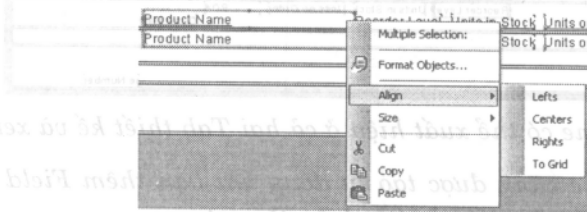
Khi di chuyển GuideLine Caret, tất cả các đối tượng gắn trong nó sẽ di chuyển theo. Để di chuyển GuideLine trong báo biểu Report, bạn thực hiện các bước sau:

1. Tìm vị trí GuideLine mà bạn muốn di chuyển và kéo Caret trượt trên thước.
2. Để loại bỏ tất cả GuideLine, nhấp phải chuột trên thanh thước và chọn mục **Remove All** từ ShortCut Menu.

4.3.2. Di chuyển và canh lề nhiều đối tượng

Bạn cũng có thể sử dụng GuideLine để di chuyển hoặc canh lề nhiều đối tượng cùng lúc. Để sử dụng được kỹ thuật này, bạn thực hiện các bước sau:

1. Trước tiên, định vị các đối tượng mà bạn muốn canh lề trong báo biểu Report, hãy chọn tất cả chúng. Bạn thực hiện việc chọn các đối tượng đó bằng cách kéo khung bao trùm qua các đối tượng hoặc nhấp vào từng đối tượng trong khi vẫn nhấn giữ phím **Shift** hoặc **Ctrl**.
2. Sau khi chọn tất cả các đối tượng mà bạn muốn canh lề, nhấp phải chuột lên trên một trong các đối tượng đó và từ ShortCut Menu xuất hiện, chọn **Align** (như hình 3.10).



Hình 3.10: Tùy biến canh lề trên ShortCut Menu.

Chọn một trong các tùy chọn canh lề cho Field của bạn và sau đó nhấp vào bất cứ nơi nào ngoài các Field đã chọn để hoàn tất.

4.3.3. Thước đo, GuideLines và khung lưới Grid

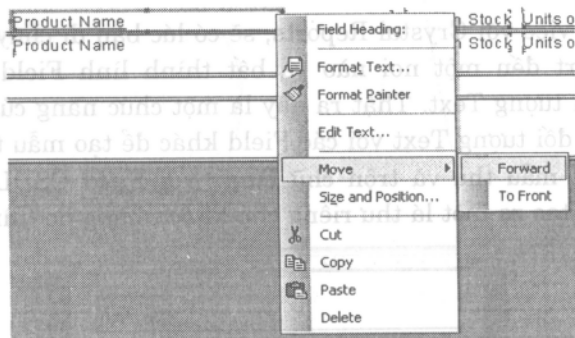
Hầu hết các tùy biến giao diện người dùng điều khiển sắp xếp trên báo biểu Report có thể thực hiện bằng cách nhấp phải chuột vào một chỗ trống trên báo biểu. Một Menu ShortCut sẽ mở ra, cho phép bạn kiểm soát diện mạo của thanh thước kẻ, đường định hướng GuideLines và khung lưới Grid xuất hiện trong Tab Design và Preview.

Thanh thước kẻ (Rule) được thiết kế để bạn đo các số đo của báo biểu, còn khung lưới Grid được sử dụng để kiểm soát vị trí các thành phần trên trang báo biểu.

4.4. Điều khiển các lớp (Layer)

Crystal Reports sử dụng các phân lớp để đặt một đối tượng lên trên một đối tượng khác. Để điều khiển các lớp đặt đối tượng bạn thực hiện các bước sau:

1. Định vị đối tượng trên báo biểu Report, nhấp phải chuột vào chúng.
2. Từ menu xuất hiện (hình 3.11), chọn mục tùy biến để điều khiển nơi mà đối tượng này được đặt đè lên đối tượng khác.



Hình 3.11: Điều khiển các lớp đặt đè đối tượng.

4.5. Đối tượng Text

Các đối tượng Text cho phép bạn đánh văn bản trực tiếp trên báo biểu. Đối tượng Text có thể kết hợp với các Field CSDL làm tiêu đề hoặc đóng vai trò như một đoạn văn bản được định dạng. Những nội dung văn bản được định dạng trước như RTF (Rich Text Format) hay HTML (HyperText Markup Language) đều có thể chèn trực tiếp vào đối tượng Text. Các đối tượng Text được dùng cho tiêu đề báo biểu Report, Field hoặc dùng trực tiếp như một nội dung mô tả.

4.5.1. Làm việc với đối tượng Text

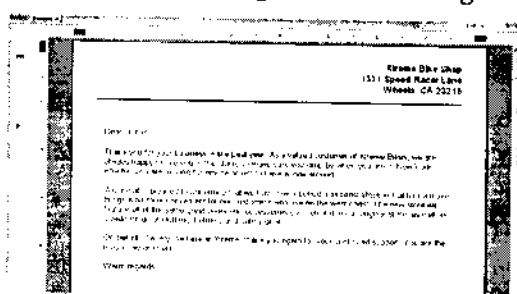
Các đối tượng Text có thể được thêm vào bất cứ nơi nào trong báo biểu Report thông qua Menu **Insert**. Khi bạn nhấp vào **Insert > Text Object**, một biểu tượng dấu chéo sẽ xuất hiện, bạn hãy dùng nó để vẽ một hộp thoại Text trong báo biểu Report. Crystal Reports sẽ chuyển sang chế độ chỉnh sửa (Edit mode) và đặt vào đối tượng Text con nháy sẵn sàng cho bạn bắt đầu nhập văn bản. Khi đã hoàn tất công đoạn chỉnh sửa Text, bạn nhấp vào bất cứ nơi nào ngoài vùng đối tượng Text để thoát khỏi chế độ chỉnh sửa.

Khi thêm vào báo biểu Report, các đối tượng Text cũng được xử lý giống như các đối tượng Field; chúng có thể di chuyển, thay đổi kích thước, định dạng... Nếu cần thay đổi nội dung Text thì bạn nhấp vào đối tượng, có thể nhấp 2 lần vào đối tượng để đặt nó trở lại chế độ chỉnh sửa hoặc bạn có thể nhấp phải chuột vào đối tượng và chọn **Edit Text** từ menu tắt.

Đối tượng Text cũng cho bạn thiết lập định dạng phân đoạn, thụt lề, nhảy dòng trống, nhập khoảng trắng, ngắt đoạn.

4.5.2. Kết hợp đối tượng Text với các Field khác

Khi làm việc với Crystal Reports, sẽ có lúc bạn di chuyển Field trong báo biểu Report đến một nơi nào và bất thành linh Field được kết hợp (merge) với đối tượng Text. Thật ra đây là một chức năng của Crystal. Bạn có thể kết hợp đối tượng Text với các Field khác để tạo mẫu thư. Ví dụ, bạn có thể tạo một mẫu thư và trộn chữ **Dear** với Field CSDL chứa tên của khách hàng để tạo ra một lá thư riêng cho khách hàng đó (hình 3.12).

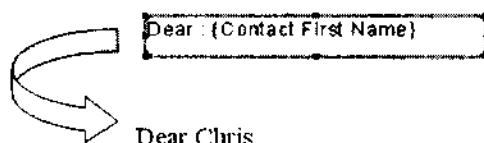


Hình 3.12: Kết hợp Field và đối tượng Text.

Để kết hợp một Field với đối tượng Text bạn thực hiện các bước sau:

1. Mở Crystal Reports và mở file báo biểu Report mẫu Blankletter.RPT từ CD-ROM.
2. Nhấp vào Tab Design để chuyển sang chế độ thiết kế.
3. Để thêm đối tượng Text cho lời chào khách hàng (**Dear XXX**), nhấp vào **Insert > Text Object** trên menu.
4. Sử dụng chuột để vẽ đối tượng Text đó lên phía trên phân đoạn chi tiết và sau đó nhập chữ **Dear**.
5. Tiếp theo, nhấp **View > Field Explorer** để mở danh sách các giá trị Field trong CSDL.
6. Mở rộng folder chứa các Field CSDL ra và chọn bảng dữ liệu **Customer**.
7. Kéo Field tên **Contact First Name** từ Field Explorer vào đối tượng Text vừa tạo lúc này. Con trỏ chuột sẽ cho bạn thấy điểm lỏng vào. Thả Field đó vào bên trong đối tượng Text vừa tạo.

8. Nhấp vào Tab Preview để xem lại báo biểu Report. Khi duyệt qua các trang, bạn sẽ thấy mỗi trang có lời chào theo tên khách hàng khác nhau như hình 3.13.



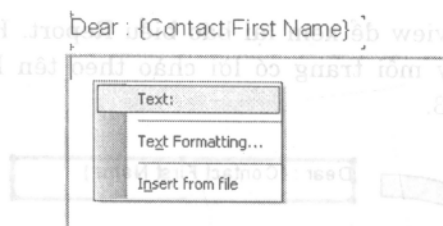
Hình 3.13: Các đối tượng Field có thể sử dụng kết hợp với Text.

4.5.3. Chèn văn bản Text đã định dạng

Khi làm việc với các đối tượng Text và các mẫu thư, báo cáo, bạn có thể không muốn nhập tất cả các văn bản trực tiếp trong Crystal Reports. Crystal Reports cũng không có khả năng kiểm tra lỗi chính tả hoặc ngữ pháp văn bản và đó là một điều khó khăn để nhập vào và định dạng một số lượng lớn văn bản trực tiếp cho đối tượng Text. Để giúp bạn, Crystal Reports cho phép sử dụng phần mềm ứng dụng xử lý **Word** tạo ra mẫu thư, mẫu chữ và sau đó mang nội dung Text hoàn chỉnh vào Crystal Reports. Ví dụ, để tạo phần nội dung bức thư bạn thực hiện qua các bước sau:

1. Mở Crystal Reports và báo biểu Report mà chúng ta đang làm việc ở phần trên.
2. Nhấp vào Tab Design để chuyển sang chế độ thiết kế báo biểu.
3. Để thêm đối tượng Text cho phần thân của bức thư bạn chọn từ menu **Insert > Text Object**.
4. Sử dụng chuột để vẽ một đối tượng Text lên trên phân đoạn chi tiết phía dưới lời chào.
5. Nhấp chuột phải vào đối tượng Text bạn vừa tạo và chọn **Insert from File** (hình 3.14), mục chọn này cho phép bạn thêm một file văn bản dạng **RTF** hoặc **HTML** vào trong đối tượng Text. Trên đĩa CD-ROM đã chứa sẵn một file dạng RTF tên là **Lettertext.RTF**. Hãy tìm và đưa file này vào đối tượng Text của bạn.

Ghi chú: Nếu muốn tăng thêm không gian, bạn có thể thay đổi kích thước trang. Hãy chọn **File > Page Setup** để truy xuất thiết lập cho kích thước trang, định hướng trang, lề và các tùy biến in khác.



Hình 3.14: Bạn có thể thêm nội dung đã định dạng trước vào đối tượng Text.

5. CÁCH HOẠT ĐỘNG CỦA CÁC PHÂN ĐOẠN (SECTION)

Crystal Reports có thể được thiết kế theo từng phân đoạn từ trên xuống. Trong chương 2 chúng ta đã lướt sơ qua các phân đoạn cơ bản để tạo nên một báo biểu trong Crystal Reports. Mặc định mỗi báo biểu Report bạn tạo có một Report Header và Report Footer dành cho báo biểu, Page Header và Page Footer dành cho trang và một phân đoạn chi tiết gọi là Detail Section (bạn cũng có thể có Group Header và Group Footer dành cho nhóm nếu bạn thêm nhóm Group vào báo biểu Report).

Với một báo biểu Report đơn giản, bạn chỉ cần những phân đoạn cơ bản đã nêu. Tuy nhiên, Crystal Reports cho phép bạn tạo nhiều phân đoạn và thiết lập thuộc tính cho từng phân đoạn riêng biệt để trợ giúp các định dạng tinh tế mà bạn có thể kiểm soát đối với các báo biểu Report phức tạp.

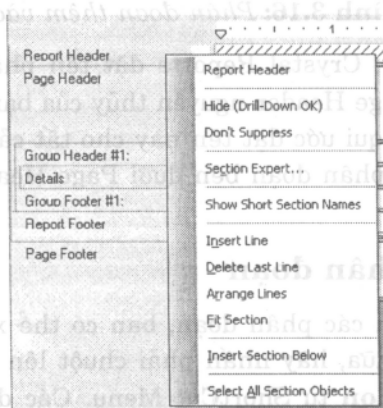
Một ví dụ sử dụng nhiều phân đoạn "Multiple section" đó là tạo mẫu thư cho công ty và muốn trình bày 2 địa chỉ khác nhau trên thư (một cho văn phòng chính và một cho văn phòng đại diện). Việc sử dụng đa phân đoạn giúp bạn có thể tạo vùng Page Header và sử dụng định dạng theo điều kiện để trình bày chính xác Header cho mỗi trang, căn cứ trên địa chỉ của khách hàng.

Để thực hiện kiểu báo biểu Report này, chúng ta cần thêm hai phân đoạn Page Header (Page Header "A" và Page Header "B") sau đó áp dụng điều kiện định dạng nào đó để quyết định chọn Page Header sẽ hiển thị. Định dạng theo điều kiện cho phép bạn tạo một công thức và nếu công thức đó là đúng thì định dạng sẽ được áp dụng.

Ghi chú: Nếu bạn chưa bao giờ làm việc với công thức Crystal Reports trước đây, chúng ta sẽ tìm hiểu trong chương 5.

Trước khi đi vào chi tiết tìm hiểu cách sử dụng các phân đoạn, chúng ta cần học một số thao tác cơ bản như thêm, xóa bỏ và trộn các phân đoạn với nhau.

Khi làm việc trong Tab Design của Crystal Reports, mỗi phân đoạn có vùng kích thước chiếm đầy bên cạnh trái của môi trường thiết kế. Nếu bạn nhấp phải chuột trên vùng trái này, một menu sẽ xuất hiện như hình 3.15.



Hình 3.15: Mỗi phân đoạn có thể được chỉnh sửa thông qua Design Tab.

Menu này cung cấp cách dễ dàng và nhanh chóng để làm việc với các phân đoạn khác nhau trong báo biểu Report. Nó là nơi mà bạn sẽ thực hiện các thao tác xử lý phân đoạn (chèn, xóa, trộn...).

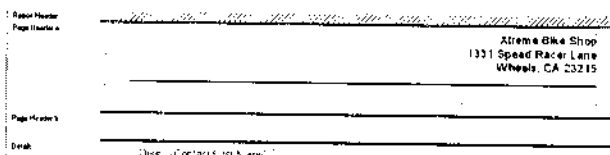
Ghi chú: Tab Preview cũng trình bày tên các phân đoạn nhưng mặc định chúng được trình bày dạng rút gọn (PH thay cho Page Header, GH1 thay cho Group Header #1...).

5.1. Thêm một phân đoạn mới

Trong ví dụ trước, bạn đã xem qua trường hợp tạo hai phân đoạn Page Header khác trong báo biểu Report để trình bày 2 giá trị trả về của địa chỉ cho mẫu thư báo biểu Report bạn tạo. Để có thể thêm một phân đoạn Page Header mới vào báo biểu, bạn thực hiện các bước sau:

1. Mở Crystal Reports và báo biểu Form Letter chúng ta đã làm ở những phần trước.
2. Chuyển sang Tab Design.
3. Tại vị trí phân đoạn Page Header, nhấp phải chuột lên nó và chọn **Insert Section Below**.

Crystal Reports thêm vào một phân đoạn ngay phía dưới Page Header trước đó như hình 3.16.



Hình 3.16: Phân đoạn thêm vào.

Bạn chú ý rằng, Crystal Reports đặt tên phân đoạn mới thêm là **Page Header B** và Page Header nguyên thủy của bạn là **Page Header A**. Crystal Reports dùng quy ước đặt tên này cho tất cả các phân đoạn khác, như nếu bạn tạo một phân đoạn bên dưới Page Header B, nó sẽ có nhãn tên là Page Header C.

5.2. Xóa một phân đoạn

Khi làm việc với các phân đoạn, bạn có thể xóa bất cứ phân đoạn nào không dùng đến nữa, hãy nhấn phải chuột lên phân đoạn muốn xóa và chọn **Delete Section** từ ShortCut Menu. Các đối tượng mà bạn đặt trong phân đoạn đó cũng sẽ bị xóa bỏ. Nếu muốn giữ các đối tượng lại, bạn phải di chuyển chúng ra khỏi phân đoạn trước khi xóa.

Ghi chú: Bạn sẽ không có khả năng xóa các phân đoạn sau của báo biểu Report bằng cách xóa trên: Report Header/Footer, Page Header/Footer, Group Header/Footer và phân đoạn chi tiết Details. Bạn có thể xóa một Group Header/Footer bằng cách xóa đi nhóm mà chúng liên hệ.

5.3. Thay đổi kích thước phân đoạn

Một kỹ thuật hữu ích khác khi làm việc với phân đoạn là thay đổi kích thước của nó. Phân đoạn có thể thay đổi kích thước để có không gian hiển thị thông tin nhiều hơn, nhưng chúng không thể lớn hơn khổ trang in. Để thay đổi kích thước một phân đoạn, thực hiện các bước sau:

1. Trong báo biểu Form Letter, định vị phân đoạn Page Header B trong Tab Design của báo biểu.
2. Ở phía dưới của mỗi phân đoạn là một đường phân cách. Di chuyển trỏ chuột đến đường phân cách đó cho đến khi con trỏ thay đổi thành hình mũi tên 2 đầu.

3. Sử dụng chuột, bạn có thể kéo phía dưới đường viền của phân đoạn lên hoặc xuống để thay đổi kích thước.

Ghi chú: *Đừng quên rằng, bạn luôn có thể dùng **Ctrl+Z** (undo) để trả về hành động đã làm trước đó nếu bạn ngẫu nhiên di chuyển một đối tượng lệch chỗ thay vì di chuyển đường khung viền.*

5.4. Tách phân đoạn

Bạn có thể tách một phân đoạn thành 2 phần. Để tách một phân đoạn bạn thực hiện các bước sau:

1. Sử dụng lại báo biểu Report mà chúng ta đã làm ở những phần trên, định vị phân đoạn bạn muốn tách và di chuyển con trỏ chuột về phía bên trái, tiến về phía dưới cùng của phân đoạn cho đến khi bạn tới giao điểm của đường thẳng thước kẻ và đường phân chia phân đoạn.
2. Con trỏ chuột của bạn sẽ trở thành một biểu tượng phân chia với một đường thẳng đơn ngay chính giữa một mũi tên lên và một mũi tên xuống.
3. Kéo chuột xuống để chia phân đoạn thành 2 phần.

5.5. Trộn các phân đoạn báo biểu

Khi bạn làm việc với nhiều phân đoạn, bạn có thể trộn các phân đoạn với nhau để xóa bớt đi hoặc đơn giản hóa phân đoạn thừa trong thiết kế báo biểu. Để trộn hai phân đoạn khác nhau, nhấp phải chuột vào phân đoạn định trộn đầu tiên và chọn **Merge Section Below** từ ShortCut Menu. Khi bạn trộn 2 phân đoạn lại, tất cả các đối tượng trong các phân đoạn đó được giữ nguyên.

Một vấn đề cần chú ý là nếu bạn trộn 2 phân đoạn cùng kiểu với nhau, ví dụ, nếu bạn có Page Header A và Page Header B, bạn có thể trộn chúng với nhau, nhưng nếu bạn thử trộn báo biểu Report Header A với Page Header A thì sẽ không thể nào trộn được.

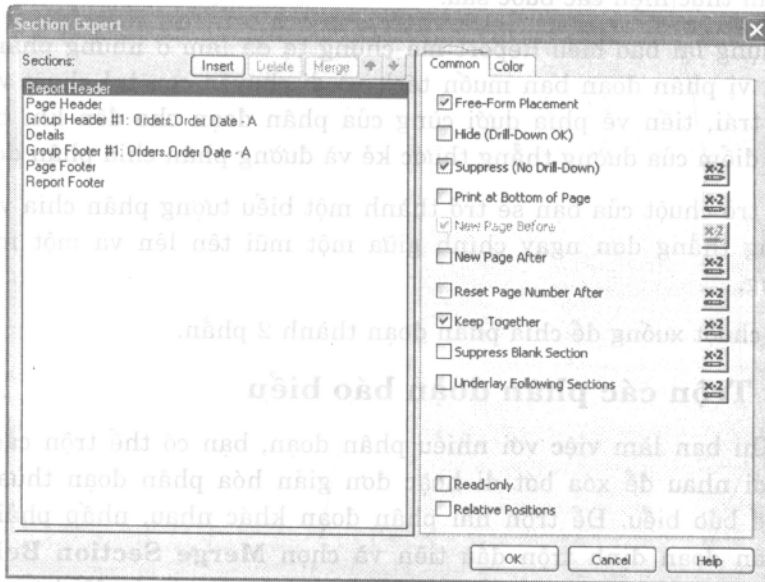
5.6. Biến đổi thứ tự các phân đoạn

Cuối cùng, bạn cũng có thể thay đổi thứ tự của các phân đoạn hiển thị trong báo biểu Report (không cần phải xóa đi để tạo lại). Để thay đổi thứ tự các phân đoạn, đơn giản là kéo và thả phân đoạn đó từ vị trí cũ sang vị trí mới. Trước tiên, bạn phải dùng con trỏ chuột giữ phân đoạn định chuyển và kéo, con trỏ chuột của bạn sẽ biến đổi thành hình bàn

tay. Khi đến được vị trí bạn cần, hãy thả phím chuột ra để đưa phân đoạn vào vị trí mới.

5.7. Định dạng các phân đoạn

Cửa sổ hộp thoại Section Expert là nơi bạn có thể hiểu được cách các phân đoạn trong báo biểu Report hoạt động chung với nhau. Bạn chọn **Report > Section Expert** để hiển thị cửa sổ này. Tất cả các phân đoạn của báo biểu Report sẽ được liệt kê như hình 3.17 cùng với tất cả các tùy biến định dạng phân đoạn.



Hình 3.17: Cửa sổ Section Expert.

Có rất nhiều tùy chọn liên quan đến tạo, sắp xếp lại và xóa các phân đoạn của báo biểu. Một số tùy chọn có thể sử dụng trực tiếp từ ShortCut Menu, nhưng sử dụng Section Expert sẽ dễ dàng hơn. Các tùy chọn này có thể tìm thấy ở phía trên của hộp thoại Section Expert, chúng gồm có:

- **Insert:** Thêm một phân đoạn mới vào báo biểu Report.
- **Delete:** Xóa một phân đoạn.
- **Merge:** Giữ lại tất cả các đối tượng có trong 2 phân đoạn và trộn chúng lại với nhau (trộn 2 phân đoạn thành 1 phân đoạn).
- **Move Up/Move Down:** Thay đổi thứ tự các phân đoạn bạn đã thêm vào báo biểu.

Mỗi phân đoạn trong báo biểu Report cũng có những tùy chọn riêng điều khiển cách ứng xử và giao diện của phân đoạn. Một danh sách các tùy chọn có sẵn dành cho phân đoạn là:

- **Free-From Placement:** Cho phép bạn đặt các đối tượng vào bất cứ nơi nào trong báo biểu Report mà không phụ thuộc vào thiết lập của khung Grid và đường định hướng (GuideLine).
- **Hide (Drill-Down OK):** Ẩn phân đoạn trong báo biểu nhưng vẫn cho phép Drill-Down xem phân đoạn này khi có yêu cầu.
- **Suppress (No Drill-Down):** Cấm hoàn toàn phân đoạn báo biểu. Bạn sẽ không thể Drill-Down để xem phân đoạn này.
- **Print at Bottom of Page:** In toàn bộ phân đoạn báo biểu Report ở cuối trang Report.
- **New Page Before:** Tạo một trang ngắt ngay trước một phân đoạn.
- **New Page After:** Tạo một trang ngắt ngay sau một phân đoạn.
- **Reset Page Number After:** Khởi động lại số thứ tự trang ngay sau một phân đoạn.
- **Keep Together:** Giữ nội dung phân đoạn liền nhau trên 1 trang.
- **Suppress Blank Section:** Ngắt phân đoạn.
- **Format with Multiple Columns:** (Chỉ thực hiện với phân đoạn chi tiết) cho phép tạo nhiều cột trong báo biểu Report.
- **Reserve a Minimum Page Footer:** (Chân trang) Dành ra mức tối thiểu phần chân trang do yêu cầu bởi máy in.
- **Color:** Kích hoạt và thiết lập màu nền cho phân đoạn.

Chúng ta sẽ lần lượt tìm hiểu cách sử dụng và ý nghĩa của những tùy chọn này.

5.8. Ẩn hoặc ngắt một phân đoạn báo biểu

Trong phần trước, bạn đã học cách đào sâu thông tin (Drill-Down) báo biểu Report bằng cách ẩn đi các phân đoạn khác nhau trong báo biểu và tổng kết báo biểu Report bằng ngắt phần chi tiết của báo biểu. Section Expert cung cấp các chức năng tương tự như vậy giúp nhanh chóng ẩn hoặc ngắt bỏ đi các phân đoạn. Từ cửa sổ Section Expert, bạn đánh dấu chọn tên phân đoạn liên quan, sau đó chọn thuộc tính **Hide** hoặc **Suppress**, nó sẽ tác động giống như khi bạn nhấn chuột phải lên

tên phân đoạn trong chế độ Design và chọn **Hide** hoặc **Suppress** từ ShortCut Menu.

5.9. Hiển thị các phân đoạn ẩn

Khi bạn quay trở lại Tab Design của báo biểu, đôi khi thật khó xác định rằng đâu là phân đoạn sẽ sử dụng và đâu là phân đoạn ẩn. Để phân biệt và hiển thị tất cả các phân đoạn ẩn trong Tab Design, bạn thực hiện theo các bước sau:

1. Nhấp vào **File > Option** và nhấp vào **Tab Layout**.
2. Từ đây, định vị phân đoạn **Design View** và nhấp vào tùy chọn **Show Hidden Sections**.
3. Click **Ok** để chấp nhận thay đổi của bạn.

Khi trở lại chế độ Design, các phân đoạn ẩn trong báo biểu Report sẽ hiển thị ngay trong Tab Design với màu xám.

5.10. In phân đoạn ở cuối trang

Một chức năng hữu ích khác là có thể in phân đoạn ở phần cuối trang. Để sử dụng kỹ thuật này bạn làm theo các bước sau:

1. Mở Crystal Reports và báo biểu Report của bạn.
2. Chuyển về chế độ Design và xác định phân đoạn bạn muốn in ở cuối mỗi trang.
3. Nhấp phải chuột vào phân đoạn và chọn **Section Expert** từ ShortCut Menu.
4. Trong Section Expert, chọn tùy chọn **Print at Bottom Page**.

5.11. Tạo ngắt trang trước hoặc sau phân đoạn

Bạn có thể tạo ngắt trang trước hoặc sau mỗi phân đoạn. Kỹ thuật này có thể được dùng với hóa đơn (tạo một đường ngắt giữa các số hóa đơn), mẫu thư (ngăn cách trang cho mỗi thư) ... Để tạo một trang ngắt trước hoặc sau một phân đoạn, bạn cần xác lập thuộc tính trong **Section Expert** cho phân đoạn là **New Page Before** hoặc **New Page After**. Khi báo biểu Report của bạn được xem ở chế độ Preview hoặc in ra giấy, nó sẽ ngắt trang như bạn đã chỉ định.

Ghi chú: Nếu bạn dùng **New Page Before**, báo biểu Report của bạn sẽ trình bày một trang trắng ngay trang đầu tiên. Đây là do phần **Report Header** xuất hiện và sau đó là đến sự kiện ngắt trang. Để giải quyết vấn đề này, hãy loại bỏ phân đoạn **Report Header**.

5.12. Đặt lại số trang sau phân đoạn

Khi làm việc với các báo cáo, hóa đơn, mẫu thư hoặc báo biểu được tạo với mục đích phân phối cho những đối tác khác nhau, bạn cần xác lập lại số trang sau mỗi phân đoạn riêng dành cho đối tác nào đó đã in xong. Trong ví dụ này, chúng ta sẽ kết hợp tùy chọn **New Page After** và **Reset Page Numbering** để tạo ngắt trang và xác lập lại số trang sau mỗi bảng báo cáo khách hàng (Customer Statement). Để thực hiện kỹ thuật này bạn làm theo các bước sau:

1. Mở Crystal Reports và mở file **Statement.RPT** từ đĩa CD-ROM.

Ghi chú: Để xác lập lại số trang sau một phân đoạn, điều trước tiên phải chắc chắn rằng bạn đã có một **Field Page Number** được thêm vào trong báo biểu Report.

2. Chọn Tab Design của báo biểu Report, nhấp **View > Field Explorer** từ menu. Mở rộng mục **Special Fields** trong cửa sổ Field Explorer, trong danh sách các Field đặc biệt này chọn Field mang tên **Page N of M Field** và kéo nó vào báo biểu Report trong phân đoạn **Group Footer #1**.
3. Nhấp phải chuột vào phân đoạn **Group Footer #1** và chọn **Section Expert** từ ShortCut Menu.
4. Sử dụng thuộc tính ở bên phải cửa sổ **Section Expert**, chọn **New Page After** và **Reset Page Number After**.
5. Nhấp **OK** để trở lại báo biểu.

Khi xem báo biểu ở chế độ Preview, bạn sẽ thấy số trang hoặc **Field Page N of M** bạn vừa thêm vào sẽ quay trở về số đếm đầu tiên sau mỗi phân đoạn.

5.13. Loại bỏ các phân đoạn trống

Bạn có thể loại bỏ những phân đoạn trống để thu gọn báo biểu Report. Kỹ thuật này thường xuyên được sử dụng khi làm việc với trường họ tên và địa chỉ. Bạn cần tạo 2 phân đoạn khác nhau cho 2 dòng địa chỉ (dòng địa chỉ 1 và dòng địa chỉ 2 như hình 3.18) và xác lập tùy chọn

Suppress a Blank Section để loại bỏ không in phân đoạn khi Field địa chỉ 2 không có giá trị.

Details a	Customer Name
Details a	Address1
Details c	Address2
Details d	(City), (Region), (Postal Code), (Country)
Details e	

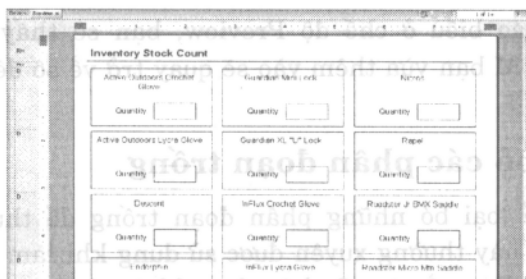
Hình 3.18: Sử dụng Section Expert không in các phân đoạn trống.

Để ngắt bỏ phân đoạn trắng trong Section Expert ở bên phải của sổ, bạn hãy chọn mục **Suppress if Blank**.

5.14. Tạo báo biểu nhiều cột

Cho đến thời điểm này, tất cả báo biểu mà chúng ta thiết kế đều ở dạng 1 cột hay còn gọi là single-column layout; nói cách khác, tất cả các Field và thành phần của báo biểu Report chỉ là dạng danh sách sổ xuống. Thông qua một số định dạng phân đoạn riêng, báo biểu Report có thể tạo ra các cột nhiều dòng, cơ chế này cho phép bạn tạo báo biểu Report linh động như danh sách trang vàng điện thoại, các danh sách, danh sách liên lạc hay những định dạng đòi hỏi thông tin hiển thị trọn trong một vùng nào đó.

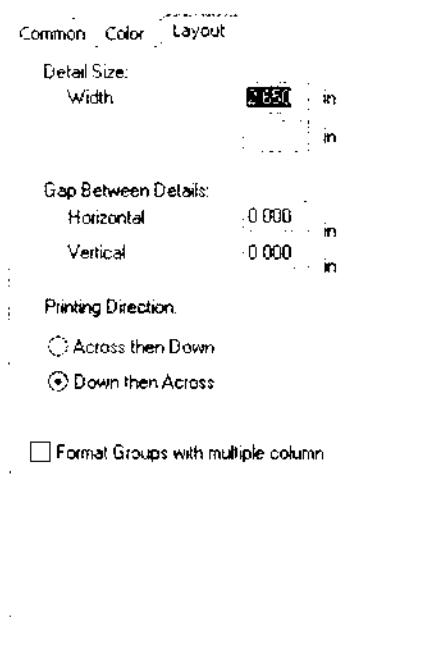
Để tạo báo biểu dạng nhiều cột, trước hết hãy tạo Report theo dạng thông thường bằng **Standard Report Expert**, thêm vào các Field bạn muốn kết xuất trong báo biểu Report cũng như các nhóm hoặc các Field tổng kết. Sau khi hoàn tất, xem kết quả ở dạng Preview và cũng là thời điểm để trình bày báo biểu ở dạng cột. Trong ví dụ này, chúng ta sẽ tạo báo biểu Report nhiều cột để hiển thị danh sách các thẻ kho. Chúng ta sẽ có một danh sách các sản phẩm chạy dọc tạo thành 3 cột, mỗi vùng thông tin là một khung văn bản hiển thị tên sản phẩm và số lượng tồn kho như hình 3.19.



Hình 3.19: Bạn có thể định dạng báo biểu thành nhiều cột.

Để tạo báo biểu Report đa cột này, bạn thực hiện theo các bước sau:

1. Mở Crystal Reports và mở file **Stocklist.RPT** từ đĩa CD-ROM.
2. Chuyển qua chế độ Design của báo biểu.
3. Nhấp phải chuột vào phân đoạn chi tiết của báo biểu Report và chọn Section Expert từ ShortCut Menu.
4. Từ danh sách tùy chọn bên cạnh phải của cửa sổ Section Expert, chọn **Format with Multiple Columns**.
5. Sau khi chọn tùy chọn này, Tab Layout sẽ xuất hiện ở phía trên cùng của danh sách tùy chọn. Nhấp vào Tab này để mở hộp thoại (hình 3.20).



Hình 3.20: Tab Layout sẽ không hiển thị trong Section Expert nếu bạn không chọn mục Format with Multiple Columns.

Điều trước tiên bạn cần làm là chọn chiều rộng cho cột, cũng như khoảng trống nằm ngang và thẳng đứng giữa mỗi vùng mẫu tin kết xuất. Ở báo biểu Report này, nhập chiều rộng bằng 2 Inches.

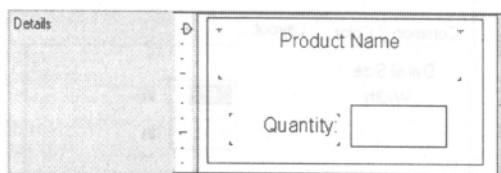
Trong hộp thoại này, bạn cũng có thể chọn hướng in bằng cách sử dụng nút Radio, *Across then Down* (ngang và hướng xuống) hoặc hướng xuống rồi sang ngang (mặc định).

Ghi chú: Nếu bạn cũng muốn định dạng cho các nhóm kết xuất nhiều cột thì hãy nhấp vào **Checkbox** ở cuối hộp thoại **Format Groups with multiple column**.

Để hoàn tất bạn nhấp vào **OK** để chấp nhận việc thay đổi và trở lại chế độ Design hoặc Preview của báo biểu Report.

Nhấp vào **Insert > Box** để vẽ hộp khung viền cho vùng thông tin hiển thị và điều chỉnh các Field của bạn lại cho vừa.

Design view của báo biểu Report sẽ hiển thị một đường phân cách xám (hình 3.21) cho biết kích thước của cột bạn đã định. Bạn sử dụng đường thẳng này như đường điều hướng để sắp xếp lại các Field báo biểu.



Hình 3.21: Đường phân cách cho biết kích thước của cột.

Nếu cần thay đổi lại thông số, bạn vào **Report > Section Expert** và chọn phân đoạn chi tiết, thực hiện lại các bước ta vừa làm trước đó.

Ghi chú: Ngoài việc kẻ khung, bạn cũng có thể vẽ các đường thẳng trên báo biểu Report bằng cách chọn **Insert > Line**.

6. THÊM NHÓM (GROUP) VÀO BÁO BIỂU

Cho đến lúc này, chúng ta đã xem qua các báo biểu Report đơn giản với cột, dòng dữ liệu và thỉnh thoảng (như trong ví dụ Form Letter chúng ta đã sử dụng trước đây) chúng ta cũng có sử dụng nhóm. Nhóm đơn giản là tập hợp các mẫu tin liên quan với nhau về một tính chất nào đó.

Khi sử dụng trong báo biểu Report, các Group cho phép bạn đặt các mẫu tin liên quan với nhau thành nhóm để phân tích thông tin. Ví dụ, nếu bạn tạo một báo biểu Report bán hàng, bạn có thể muốn nhóm các mẫu tin khách hàng theo từng quốc gia (hình 3.22), hoặc nhóm danh sách các đơn hàng theo từng khách hàng.

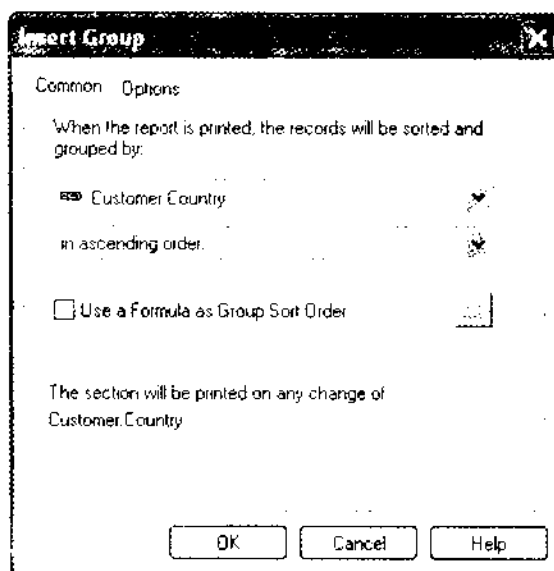
Đối với mỗi nhóm Group mà bạn tạo trong báo biểu, Crystal Reports sẽ tạo một cặp Group Header và Group Footer (hình 3.23). Đây là nơi bạn có thể đặt tiêu đề cho nhóm hoặc các giá trị tổng kết cho nhóm dữ liệu.

Hình 3.22: Báo biểu danh sách hàng bán nhóm theo từng quốc gia.

	Country
Group Header #1:	Group #1 Name
Details	Country
Group Footer #1:	
Report Footer	Grand Total:
Page Footer	

Nhóm có thể dựa trên các Field CSDL, các Field tham số, các Field công thức hoặc câu lệnh SQL xuất hiện trong báo biểu Report. Để thêm một nhóm vào báo biểu Report, bạn thực hiện theo các bước sau:

1. Mở Crystal Reports và mở file báo biểu Report **Customerlisting.RPT** trên CD-ROM.
2. Chọn **Insert > Group**. Một hộp thoại xuất hiện như hình 3.24, chọn Field để sắp xếp và nhóm các mẫu tin trong báo biểu.
3. Hãy chọn Field **Country** từ danh sách Drop-Down và chọn **In ascending order** cho phần sắp xếp thứ tự tăng dần.
4. Khi hoàn tất thiết lập tùy chọn nhóm, nhấp **OK** để thêm nhóm vào báo biểu Report. Lưu báo biểu Report này lại thành file **Customercountry.RPT** (chúng ta sẽ sử dụng lại sau trong chương này).



Hình 3.24: Các thiết lập nhóm.

Các Group sẽ hiển thị ngay trong cây phân nhóm (Group Tree) bên phía trái của trang, Group Header và Group Footer cùng với tên Group cũng sẽ được thêm vào báo biểu Report.

Ghi chú: Để xóa một Group, bạn chuyển qua Tab Design, nhấp phải chuột để chọn Group Header hoặc Group Footer và chọn **Delete Group** từ ShortCut Menu.

6.1. Tùy chọn định dạng nhóm

Với bất kỳ nhóm nào bạn cũng có thể điều khiển cách hiển thị, sắp xếp nội dung của nhóm trong báo biểu. Hãy sử dụng những kỹ thuật trong phần tiếp theo sau đây để định dạng nhóm.

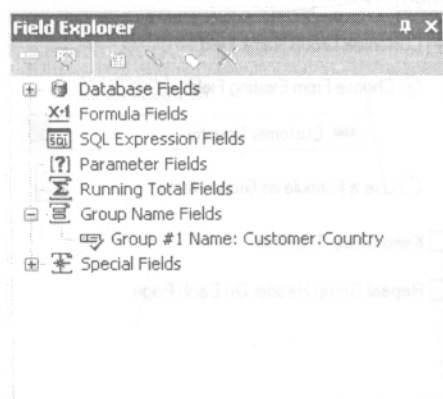
6.1.1. Thêm các tên nhóm

Tên nhóm là nhãn do Crystal Reports tự động thêm vào mỗi khi bạn tạo nhóm. Chúng thường xuất hiện ở cả hai phân đoạn Group Header và Group Footer.

Sẽ có trường hợp bạn muốn thêm tên nhóm thủ công và Crystal Reports cho phép bạn thực hiện điều này. Để thêm tên nhóm bạn thực hiện theo các bước sau:

1. Để dễ xem chuyển qua chế độ Design của báo biểu.

- Sau đó nhấp vào **View > Field Explorer**. Cửa sổ Field Explorer sẽ hiển thị (hình 3.25).



Hình 3.25: Field Explorer có thể sử dụng để thêm các tên Group vào báo biểu Report.

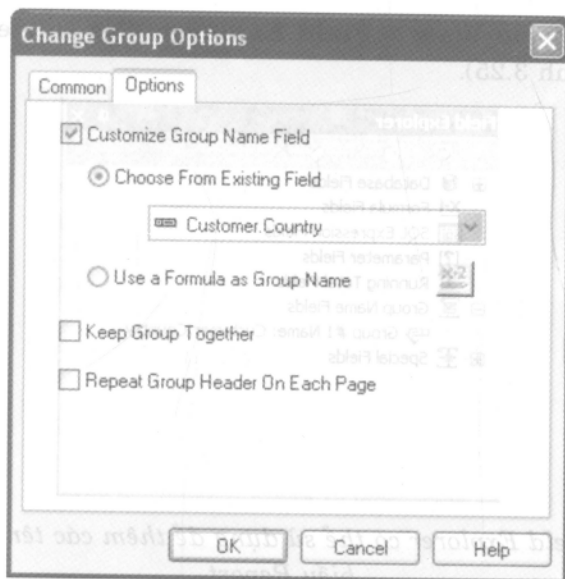
- Nhấp vào biểu tượng dấu cộng mục **Group Name Fields**. Một danh sách tất cả các tên nhóm của báo biểu Report sẽ xuất hiện.
- Chọn Field mà bạn muốn thêm vào trong báo biểu Report và nhấn phím **Enter**. Field sẽ được gắn vào đầu con trỏ chuột và khi di chuyển chuột vào trang báo biểu Report, bạn sẽ thấy đường viền bao quanh Field.
- Di chuyển chuột đến vị trí muốn đặt Group Name Field, nhấp chuột để thêm Field vào báo biểu. Khi xem báo biểu ở chế độ Preview, Field này sẽ được thay thế bằng tên của nhóm mà nó đại diện.

6.1.2. Tùy biến Group Name Field

Các Group Name Field có thể tùy biến theo nhiều cách. Bạn có thể truy xuất các tùy chọn để tùy biến tên nhóm khi lần đầu tiên chèn nhóm mới vào báo biểu hoặc khi bạn thay đổi thuộc tính của nhóm.

Ghi chú: Để thay đổi một nhóm đã tồn tại, bạn có thể sử dụng **Group Expert** bằng cách chọn mục **Report > Group Expert**.

Như trong hình 3.26, bạn có thể chọn một tên nhóm cho Field đang có bằng cách chọn mục **Choose from Existing Field** sau đó chọn tên Field từ danh sách. Một ví dụ phổ biến để minh họa là khi bạn muốn sử dụng nhóm Field **Company ID** nhưng lại muốn hiển thị **Company Name**.



Hình 3.26: Chọn Group Name Field.

Bạn cũng có thể chọn một tên nhóm dựa vào công thức (formula) bằng cách nhấp chọn **Use a Formula As Group Name** hay nhập vào công thức sử dụng nút **X+2**. Cửa sổ trình soạn thảo công thức của Crystal Reports sẽ được mở ra cho phép bạn nhập công thức trả về giá trị cho tên nhóm. Một ví dụ minh họa đó là, bạn muốn sử dụng công thức nhóm mã các đại lý bán hàng (**sales rep code**) nhưng lại muốn hiển thị tên của các đại lý bán hàng (**Sales rep's name**). Công thức thiết lập trong trường hợp này sẽ như sau:

```
If {Customer.RepNo} = 112 then "Nathan ' S Customers", else
If {Customer.RepNo} = 234 then "Kelly ' S Customers" else
If {Customer.RepNo} = 258 then "Jane ' S Customers" else
"Other Customers"
```

Sau khi bạn đã nhập công thức tùy chọn cho tên nhóm và thoát khỏi trình soạn thảo, bạn sẽ thấy nút **X+2** sẽ chuyển từ màu xanh biển sang màu đỏ và biểu tượng cây viết được di chuyển từ nằm ngang thành nằm xiên để cho biết rằng bạn đã nhập công thức vào (hình 3.27).

☒ Use a Formula as Group Name

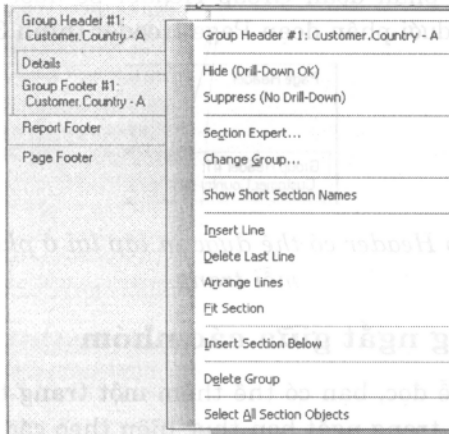


Hình 3.27: Nút X+2 thay đổi từ màu xanh thành màu đỏ và biểu tượng cây viết thay đổi vị trí cho biết công thức đã được nhập.

6.1.3. Thay đổi tiêu chí của nhóm

Sau khi nhóm được chèn vào, bạn có thể thay đổi tiêu chí của nhóm. Trong chế độ Design, chọn Group Header hoặc Group Footer của nhóm mà bạn muốn thay đổi tiêu chí.

Nhấp phải chuột vào vùng màu xám của Group Header hoặc Group Footer sẽ xuất hiện một menu (hình 3.28), từ menu này hãy chọn **Change Group**.



Hình 3.28: Mục chọn hiển thị hộp thoại tùy chọn nhóm.

Bạn có thể thay đổi các tùy chọn cho nhóm trong hộp thoại Change Group này. Nhấp **OK** để chấp nhận việc thay đổi của bạn. Các thay đổi sẽ có tác dụng ngay lập tức trong báo biểu Report Design hoặc của sổ Preview.

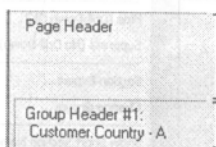
6.1.4. Giữ các nhóm liên nhau giữa các trang

Tùy chọn này giúp cho việc giữ các nhóm liên lạc với nhau trong một trang. Crystal Report sẽ cố gắng hiển thị toàn bộ nhóm trên cùng một trang. Với chức năng này, bạn có thể chọn mục **Keep Group Together** từ hộp thoại **Insert Group** hoặc hộp thoại **Change Group Option**. Nhấp **OK** để chấp nhận việc thay đổi. Khi báo biểu Report của bạn được xem ở chế độ Preview hoặc in ra, Crystal Reports sẽ cố gắng di chuyển tất cả các nhóm mẫu tin vào một trang đơn.

Ghi chú: Trong trường hợp Crystal Reports không thể điều chỉnh tất cả các mẫu tin của nhóm nằm vừa trên cùng trang, nó sẽ in theo cách thông thường.

6.1.5. Lập lại tiêu đề nhóm ở mỗi trang

Đối với các báo biểu Report dài, tiêu đề nhóm Group Header có thể phải xuất hiện trên mỗi trang nhằm giúp người đọc báo biểu có thể nhận dạng mỗi nhóm (do Group Header và Group Footer có thể cách nhau từ 10 đến 20 trang). Để lập lại tiêu đề nhóm Group Header trên mỗi trang, bạn chọn mục **Repeat a Group Header** khi lần đầu thêm một nhóm mới hoặc khi bạn thay đổi thuộc tính nhóm. Khi báo biểu Report được xem ở chế độ Preview hoặc in ra, phân đoạn Group Header sẽ được in ở trên cùng của mỗi trang ngay bên dưới phân đoạn Page Header (hình 3.29).



Hình 3.29: Group Header có thể được in lập lại ở phần phía trên của mỗi trang.

6.2. Tạo trang ngắt giữa các nhóm

Để báo biểu dễ đọc, bạn có thể thêm một trang trắng ngắt giữa các nhóm. Để thêm một trang ngắt bạn thực hiện theo các bước sau:

1. Nhấp vào **Format > Section** để mở Section Expert.
2. Định vị trí Group Footer của nhóm mà bạn muốn sử dụng làm nơi ngắt trang. Đánh dấu nhóm bằng cách nhấp chọn chúng.
3. Từ tùy chọn xuất hiện trên cạnh phải của trang, chọn **New Page After**.
4. Nhấp **OK** để trở lại báo biểu Report của bạn.

Ghi chú: Bạn cũng có thể làm theo cách khác là chọn Group Header và chọn từ Menu ShortCut mục **New page Before**.

Báo biểu Report của bạn sẽ có một trang ngắt ngay bên dưới mỗi nhóm giúp người xem dễ đọc hơn.

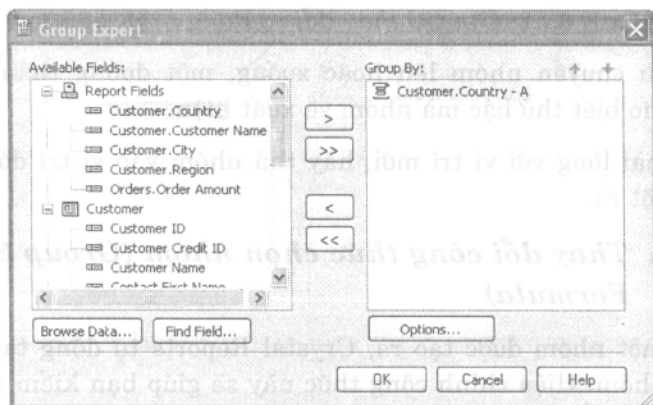
6.3. Thay đổi nhóm

Khi đã thêm nhóm vào trong báo biểu, bạn có thể thay đổi các thuộc tính của nhóm mà không cần phải xóa bỏ hoặc tạo mới chúng. Bạn có thể muốn thay đổi các Field được nhóm (Group Field) hoặc thứ tự sắp xếp (sort order) hay chỉ muốn xem những tùy chọn mà bạn đã thiết lập

cho nhóm. Hộp thoại Change Group Expert sẽ giúp bạn thực hiện những công việc này.

6.3.1. Sử dụng Change Group Expert

Change Group Expert có thể được sử dụng cho việc thay đổi thiết lập các nhóm và các tùy chọn nhóm. Bạn chọn **Report > Change Group Expert**, chọn nhóm mà bạn muốn thay đổi và sau đó chọn nút **Options**.



Hình 3.30: Bạn có thể dùng Change Group Expert để thay đổi các thiết lập nhóm.

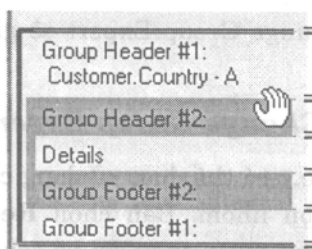
Hộp thoại Change Group Options sẽ xuất hiện, cho phép bạn thay đổi Field dùng để nhóm (Group Field), sắp xếp (sort order) ... Nhấp **OK** để chấp nhận thay đổi của bạn. Thay đổi sẽ tác dụng ngay lên báo biểu Report đang thiết kế.

6.3.2. Sắp xếp lại các nhóm

Khi làm việc với các nhóm, bạn rất dễ mất dấu thứ bậc của chúng (ví dụ khi thêm một nhóm **Country** bên trong một nhóm **State**). Để sắp xếp lại thứ tự các nhóm xuất hiện trong báo biểu, bạn đơn giản nắm kéo và thả các phân đoạn theo từng bước như sau:

1. Trong chế độ Design, định vị các Group Header và Group Footer trong vùng màu xám bên cạnh phải của trang.
2. Định vị các Group bạn muốn sắp xếp và di chuyển chúng lên hoặc xuống bằng cách kéo các phân đoạn tương ứng của báo biểu.

Khi bạn đã chọn một nhóm, con trỏ chuột xuất hiện dạng hình bàn tay (hình 3.31), cho biết nhóm sẵn sàng để bạn di chuyển.



Hình 3.31: Con trỏ thay đổi có dạng hình bàn tay.

Khi di chuyển nhóm lên hoặc xuống, một đường thẳng dày đậm xuất hiện cho biết thứ bậc mà nhóm sẽ xuất hiện.

Nếu hài lòng với vị trí mới, hãy thả nhóm vào vị trí đó bằng cách thả nút chuột ra.

6.3.3. Thay đổi công thức chọn nhóm (Group Selection Formula)

Khi một nhóm được tạo ra, Crystal Reports tự động tạo một công thức chọn nhóm. Hiệu chỉnh công thức này sẽ giúp bạn kiểm soát dữ liệu xuất hiện trong mỗi nhóm tốt hơn. Bạn có thể hiệu chỉnh Group Selection Formula bằng cách nhấp vào **Report > Selection Formulas > Group**. Trình soạn thảo công thức Crystal Reports Formula Editor sẽ hiển thị. Bạn có thể sử dụng trình soạn thảo này để thay đổi công thức.

Ghi chú: Xem chương 5 sẽ hướng dẫn bạn kỹ hơn về cách tạo công thức.

6.3.4. Thứ bậc nhóm (Hierarchical Grouping)

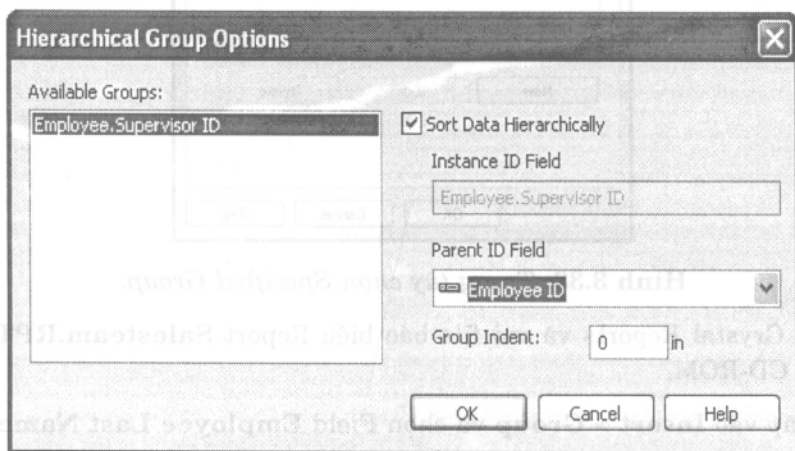
Khi làm việc với các nhóm (Group) hoặc sắp xếp (Sort), bạn đôi khi giả định theo thiết kế của CSDL. Ví dụ, để có một báo biểu Report cho biết doanh số bán toàn cầu, bạn sẽ nhóm các Field theo quốc gia (Country), vùng lãnh thổ (Region) và thành phố (City), như vậy là ta nhóm theo 3 cấp bậc.

Tuy nhiên, nếu cần nhóm dữ liệu của chỉ một Field trong bảng dữ liệu theo nhiều cấp thì sao? Một ví dụ để bạn dễ hình dung đó là, giả sử bạn có một bảng dữ liệu Employee chứa tất cả chi tiết của nhân viên làm việc trong công ty của bạn. Trong bảng này, Field ManagerID chứa mã nhân viên của người quản lý. Bạn muốn xem quan hệ phân cấp báo biểu này như thế nào thì không thể nhóm theo cách thông thường được. Vì nhóm thông thường theo ManagerID chỉ có thể liệt kê danh sách nhân

viên mà mỗi người quản lý, không cho biết ai thuộc cấp quản lý thứ bậc mấy trong sơ đồ tổ chức của công ty.

Để tạo thứ bậc khi nhóm một Field bạn thực hiện qua các bước sau:

1. Mở Crystal Reports và mở file báo biểu Report **Hierarchy.RPT** trên CD-ROM.
2. Thêm vào một nhóm bằng cách nhấp chọn **Insert > Group** và chọn Field dùng liên kết Field thứ bậc. Trong ví dụ này, chúng ta sẽ làm việc với bảng dữ liệu **Employee** và Field dùng để nhóm là mã người quản lý (**Supervisor ID**). Để hiển thị thứ bậc của những nhân viên nào làm việc cho họ, hãy chọn **Supervisor ID** cho nhóm và nhấp vào **OK**.
3. Nhấp vào **Report > Hierarchical Grouping Options** để mở hộp thoại như hình 3.32.



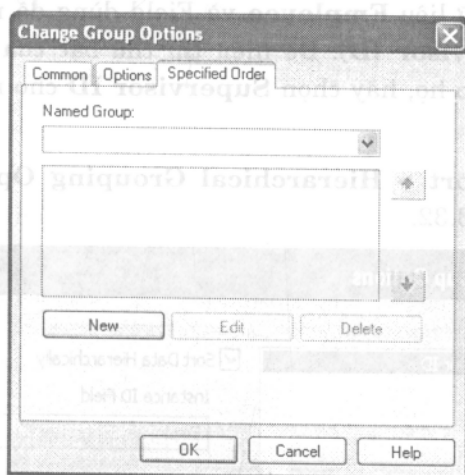
Hình 3.32: Hộp thoại tùy chọn thiết lập cấp bậc nhóm.

- Chọn tùy chọn **Sort Data Hierarchically**. Từ danh sách **Available Groups**, bạn hãy chọn nhóm mà bạn muốn nhóm theo cấp bậc.
- Chọn **Parent ID Field** là Field dùng để liên hệ cấp bậc. Trong ví dụ này, bạn chọn Field **Employee ID**.

Cuối cùng, thiết lập kích thước thụt đầu dòng cho mỗi cấp trong ô Group Indent. Nhấp vào **OK** để chấp nhận các thay đổi.

6.4. Nhóm theo chỉ định

Nhóm theo chỉ định là một chức năng mạnh, chúng cho phép bạn tạo nhóm dữ liệu dựa trên tiêu thức nào đó. Ví dụ, bạn có các khu vực bán hàng bao gồm các vùng lãnh thổ (state). Bạn có thể dùng một nhóm theo chỉ định để tạo phân cách nhóm (separate group) cho mỗi khu vực bán hàng theo cách riêng của mình (ví dụ, North Carolina và South Carolina tạo nên một nhóm khu vực bán hàng của Bob). Để tạo một nhóm chỉ định bạn thực hiện theo các bước sau:



Hình 3.33: Trang tùy chọn Specified Group.

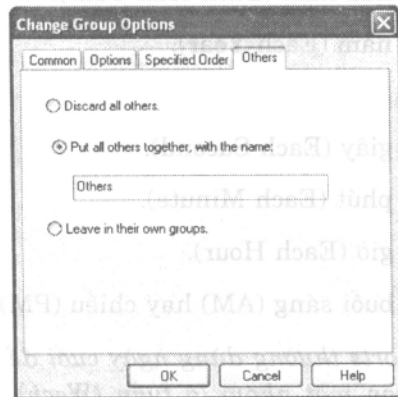
1. Mở Crystal Reports và mở file báo biểu Report **Salesteam.RPT** trên đĩa CD-ROM.
2. Nhấp vào **Insert > Group** và chọn Field **Employee Last Name**.

Trong hộp thoại **Insert Group**, thay đổi Sort order thành **In Specified order**. Ở Tab thứ hai bạn sẽ thấy nhãn Specified Order hiển thị trong hộp thoại **Insert Group** như hình 3.33.

Đối với mỗi một nhóm chỉ định muốn tạo, bạn cần định nghĩa tên nhóm (Group name) và chỉ rõ tiêu thức nhóm.

3. Gõ vào tất cả các tên nhóm (Group name) bạn muốn tạo, nhấn phím **Enter** sau mỗi nhóm để tạo danh sách của các tên nhóm (Group name).
4. Sau khi định nghĩa tất cả tên nhóm (Group name), đánh dấu chọn từng nhóm và nhấp nút Edit để chỉ định tiêu thức.

5. Để thiết lập tiêu thức cho các mẫu tin nhóm, bạn sử dụng danh sách Drop-Down Menu để chọn toán tử (operator) sử dụng và giá trị (value) dùng để nhóm.
6. Bạn có thể thêm các tiêu thức khác bằng cách nhấp vào Tab New và sử dụng các toán tử để chỉ định thêm những tiêu chí tuyển chọn mẫu tin cho nhóm.
7. Sau khi đã nhập vào nhóm, một Tab khác sẽ xuất hiện với các tùy chọn cho phép bạn xử lý mẫu tin không thỏa tiêu thức. Mặc định tất cả các phần còn lại của mẫu tin không thỏa sẽ được đặt riêng trong nhóm của chúng xác định bởi Tab mang tên Others. Bạn có thể thay đổi tên của nhóm này bằng cách gõ vào ô Others như hình 3.34. Bạn cũng có thể chọn loại bỏ tất cả các mẫu tin hoặc giữ nguyên trong nhóm của chúng.



Hình 3.34: Tùy chọn xử lý các mẫu tin khác.

Sau khi đã định nghĩa các nhóm và tiêu thức, nhấp vào **OK** để chấp nhận những thay đổi.

Các thiết lập về nhóm chỉ định của bạn sẽ có hiệu lực ngay trên báo biểu.

6.5. Nhóm các Field dạng ngày hoặc ngày-giờ

Khi thực hiện nhóm dữ liệu trên một Field kiểu ngày (Date Field) đòi hỏi bạn phải chỉ rõ cách mà ngày được nhóm. Để tạo một nhóm dựa trên cơ sở Field dạng ngày (Date Field), bạn nhấp vào **Insert > Group** sau đó chọn Field dạng kiểu ngày (Date) hoặc kiểu ngày giờ (Date-Time) (CSDL, Formula, Parameter) dùng cho việc sắp xếp và nhóm các mẫu tin trong báo biểu Report.

Crystal Reports sẽ thêm các ô tùy chọn cho phép bạn xác định khoảng cách thời gian sẽ nhóm như:

Với ngày tháng:

- Nhóm theo mỗi ngày (Each Day).
- Nhóm theo mỗi tuần (Each Week).
- Nhóm theo mỗi hai tuần (Each Two Weeks).
- Nhóm theo mỗi nửa tháng (Each Half-Month).
- Nhóm theo mỗi tháng (Each Month).
- Nhóm theo mỗi một quý (Each Quarter).
- Nhóm theo mỗi nửa năm (Each Half-Year).
- Nhóm theo mỗi năm (Each Year).

Với kiểu giờ:

- Nhóm theo mỗi giây (Each Second).
- Nhóm theo mỗi phút (Each Minute).
- Nhóm theo mỗi giờ (Each Hour).
- Nhóm theo mỗi buổi sáng (AM) hay chiều (PM).

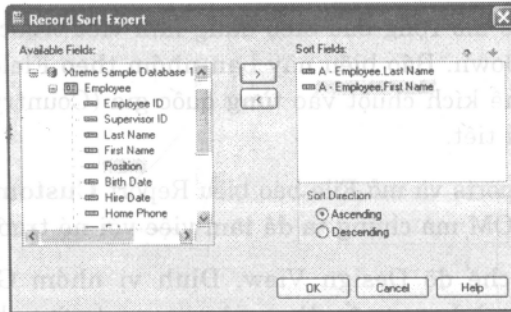
Ghi chú: Crystal Reports thường dùng ngày cuối để tạo tên nhóm. Ví dụ, nếu bạn chọn một nhóm là tuần (Week), báo biểu sẽ hiển thị ngày cuối cùng của mỗi tuần làm tên nhóm.

7. SẮP XẾP Ở CẤP MẪU TIN

Đôi lúc bạn muốn sắp xếp ngay trên mẫu tin mà không cần phải gom nhóm chúng. Bạn mở hộp thoại như hình 3.35 bằng cách chọn **Report > Record Sort Expert**.

Trong hộp thoại này, bạn di chuyển các Field bạn muốn dùng để sắp xếp từ danh sách bên trái sang danh sách bên phải bằng cách sử dụng mũi tên nằm giữa hộp thoại. Đánh dấu chọn (Highlight) Field ở bên trái, sau đó nhấp vào mũi tên bên phải để di chuyển Field sang danh sách bên phải. Bạn có thể sử dụng nút Radio bên dưới danh sách để chọn kiểu sắp xếp thứ tự tăng hay giảm dần. Mũi tên ở góc trên bên phải của hộp thoại được dùng để chỉ định thứ tự sắp xếp.

Ghi chú: Sắp xếp theo nhóm vẫn được ưu tiên hơn sắp xếp ở cấp mẫu tin. Nếu bạn có các nhóm thì những mẫu tin trong nhóm sẽ được sắp xếp theo tiêu chí của nhóm thay cho tiêu chí bạn chỉ định trong hộp thoại này.



Hình 3.35: Tùy chọn cho Record-Level Sorting.

8. DRILL-DOWN VÀ BÁO CÁO BÁO BIỂU REPORT

Việc trình diễn thông tin có ý nghĩa từ một lượng lớn dữ liệu có thể là rất khó. Nếu báo biểu Report quá dài, người sử dụng báo biểu Report có thể mệt mỏi với hàng tá dữ liệu trước khi đọc đến phần đoạn dữ liệu tổng kết có ý nghĩa. Một kỹ thuật để trình bày thông tin súc tích là dùng kỹ thuật đào sâu (Drill-Down) hoặc tổng (Summary) kết báo biểu Report.

Báo biểu Drill-Down và Summary là tương tự nhau do chúng chỉ chứa các thông tin tổng kết (Summary). Chẳng hạn, đối với một báo biểu Report bán hàng (Sales), bạn có thể muốn tổng kết người bán hàng nào bán được nhiều hàng nhất trong tháng, chúng ta chỉ việc xem con số kết quả doanh số bán đã đánh giá được, không cần xem chi tiết quá nhiều thông tin bán hàng.

Tuy nhiên, báo biểu dạng Drill-Down có thêm khả năng cho phép bạn nhấp 2 lần (double-click) vào dữ liệu tổng kết để hiển thị thông tin chi tiết của các mẫu tin hình thành nên số liệu tổng đó. Trong báo biểu dạng Drill-Down, thông tin chi tiết được ẩn đi và luôn sẵn sàng yêu cầu hiển thị. Với báo biểu Report dạng Summary thì không thể đào sâu xem tiếp thông tin chi tiết.

Mỗi kiểu báo biểu đều có cách dùng riêng. Nếu báo biểu Report mà bạn tạo cần những thông tin như “Tại sao tháng này John bán được nhiều hàng?” hoặc “Tại sao doanh số lại âm?” thì báo biểu dạng Drill-Down có thể cung cấp thông tin đầy đủ mà bạn cần.

Chúng ta sẽ mở rộng báo biểu đang làm việc trước đây để tìm hiểu tính năng Drill-Down. Báo biểu này bạn nhóm theo Field Country, người dùng sau đó có thể kích chuột vào từng quốc gia (Country) để Drill-Down xem thông tin chi tiết.

1. Mở Crystal Reports và mở File báo biểu Report **Customercountry.RPT** trên đĩa CD-ROM mà chúng ta đã làm việc với nó trước đây.
2. Chuyển sang chế độ Design View. Định vị nhóm Group Header và Group Footer mà bạn muốn làm việc (trong trường hợp này là Field Country).
3. Nhấp phải chuột (right-click) lên tiêu đề nằm trong vùng màu xám cạnh trái của trang và chọn **Hide** từ ShortCut Menu.

Báo biểu Report của bạn bây giờ sẽ hiển thị chỉ mỗi thông tin tổng kết (Summary). Để Drill-Down xem thông tin chi tiết của báo biểu bạn di chuyển chuột sang Field Group Header hoặc Group Footer, nhấp 2 lần chuột (Double-click), con trỏ chuột của bạn sẽ biến thành biểu tượng đồng hồ cát. Một Tab mới được thêm vào (cạnh Tab Design và Preview) sẽ hiển thị dữ liệu chi tiết (Drill-Down).

Ghi chú: Nếu muốn hiển thị các tiêu đề Header khi Drill-Down, bạn cần phải chọn **File > Report Options** và chọn **Show All Headers on Drill-Down**. Nhấp **OK** để chấp nhận thay đổi của bạn.

9. TÓM TẮT

Phần lớn quá trình thiết kế báo biểu Report là định dạng báo biểu biến khối lượng lớn dữ liệu thành thông tin cô đọng. Trong chương này, chúng ta đã xem qua một số kỹ thuật giúp bạn định dạng và cấu trúc báo biểu, trình diễn những thông tin có ý nghĩa và dễ đọc. Trong chương kế tiếp, chúng ta sẽ học cách tăng giá trị của báo biểu thông qua phân tích dữ liệu mà báo biểu cung cấp.

Chương 4:

PHÂN TÍCH DỮ LIỆU BÁO BIỂU

- ❖ *Tổng quan.*
- ❖ *Tổng kết dữ liệu báo biểu.*
- ❖ *Thêm Field tổng kết trong báo biểu.*
- ❖ *Phân tích dữ liệu báo biểu.*
- ❖ *Sử dụng Record Selection để lọc báo biểu.*
- ❖ *Sử dụng tham số trong báo biểu.*
- ❖ *Field tham số và Record Selection.*
- ❖ *Tóm tắt.*

1. TỔNG QUAN

Báo biểu có thể chứa số lượng lớn dữ liệu, người sử dụng báo biểu thường thấy khó có thể xem và chuyển qua hàng trăm nghìn dòng để tìm thấy thông tin họ cần.

May mắn, Crystal Reports cung cấp khả năng tổng kết và lọc dữ liệu chỉ để hiển thị thông tin cần thiết. Trong chương này chúng ta sẽ tìm hiểu kỹ thuật đó.

Trong phần đầu của chương, chúng ta sẽ học cách tổng kết dữ liệu báo biểu. Crystal Reports cung cấp sẵn cho bạn ngôn ngữ xây dựng công thức cùng các toán tử để tính tổng. Mục này chúng ta cũng sẽ học cách phân tích dữ liệu báo biểu thông qua sắp xếp, phân nhóm và tính tổng gộp để làm nổi bật thông tin mà người sử dụng có thể không nhận ra khi xem báo biểu.

Phần còn lại của chương, bạn sẽ học cách sử dụng các Field tham số, đây là những Field có thể sử dụng để lọc nội dung báo biểu chỉ giữ lại thông tin thích hợp. Ngoài việc thêm vào các Field tham số khi chạy báo biểu, một chức năng mới của Crystal Report XI là cho phép bạn ràng buộc Field tham số với các Field CSDL, cho phép người sử dụng lựa chọn

động danh sách các giá trị tham số từ một danh sách dữ liệu của bảng trong CSDL.

Nếu bạn cần tổng kết số lượng lớn dữ liệu và trình bày báo biểu một cách ngắn gọn, đầy đủ ý nghĩa, chương này sẽ là nơi chúng ta bắt đầu.

2. TỔNG KẾT DỮ LIỆU BÁO BIỂU

Một trong những cách thông dụng nhất để phân tích dữ liệu trong các báo biểu là trình bày dữ liệu dạng tổng kết. Ví dụ: bạn có thể có một báo biểu với các nhóm nhân viên bán hàng trong công ty và muốn hiển thị kết quả tổng kết doanh số đạt được của mỗi nhân viên ở cột cuối cùng như hình 4.1:

Employee Sales Summary			
Employee Name	Average Order Amount	Total Order Amount	Number of Orders
Anne Dodsworth	\$1,855.57	\$682,849.21	368
Janet Leverling	\$1,838.82	\$649,101.99	353
Margaret Peacock	\$1,858.23	\$631,799.77	340
Michael Suyama	\$1,909.68	\$710,401.48	372
Nancy Davolio	\$1,835.44	\$660,756.95	360
Robert King	\$1,876.58	\$748,755.94	399

Hình 4.1: Ví dụ báo biểu dạng tổng kết.

Bạn có thể trình bày số liệu tổng kết theo nhiều cách. Phần sau sẽ giúp bạn quyết định dạng báo biểu nào là tốt nhất.

2.1. Dạng tổng kết (Summary) so với công thức (Formula)

Khi thêm công thức tính toán vào báo biểu, bạn có hai lựa chọn: bạn có thể chèn một Field tổng kết hoặc Field tính toán theo cách riêng của mình bằng cách sử dụng công thức hay các biểu thức của SQL.

Những Field tổng kết được thiết kế để thay cho việc phải viết các công thức hoặc biểu thức SQL đối với những phép tính thông dụng như tính tổng, trung bình cộng, đếm...

Tuy nhiên, dù dễ dàng sử dụng, các Field tổng kết vẫn có những hạn chế của nó. Các Field tổng kết không linh động như những công thức do bạn tự viết, và hiện thời chỉ có 19 công thức tính tổng được

Crystal thiết kế sẵn. Những hàm công thức tính tổng này cũng bị ràng buộc theo một nhóm hoặc dòng tổng gộp của báo biểu.

Nếu bạn cần điều khiển toàn bộ cách tính toán cho ra kết quả hoặc có thể đặt kết quả công thức ở bất kỳ nơi đâu trong báo biểu, bạn cần phải học cách viết biểu thức của câu lệnh SQL và thực hiện các phép toán trên cơ sở dữ liệu máy chủ.

Dưới đây là danh sách các hàm tính tổng được Crystal cung cấp sẵn:

- **Sum:** Tính tổng nội dung của tất cả mẫu tin trong một nhóm (Group) có trường kiểu dữ liệu số như Numeric, Currency.
- **Average:** Tính trung bình cộng của tất cả mẫu tin trong một nhóm (Group) có trường kiểu dữ liệu số như Numeric, Currency (tức là, cộng tất cả các giá trị của các trường rồi chia cho tổng số).
- **Minimum:** Xác định giá trị nhỏ nhất của Field hiện có trong cơ sở dữ liệu. Các Field này phải sử dụng các kiểu dữ liệu như: Numeric, Currency, String, Date.
- **Maximum:** Xác định giá trị lớn nhất của Field hiện có trong cơ sở dữ liệu. Các Field này phải sử dụng các kiểu dữ liệu như: Numeric, Currency, String, Date.
- **Count:** Đếm những giá trị của Field hiện có trong cơ sở dữ liệu. Field này được sử dụng với tất cả các kiểu dữ liệu.
- **Distinct Count:** Đếm những giá trị của Field hiện có trong cơ sở dữ liệu, nhưng những giá trị giống nhau chỉ đếm một lần.

Ngoài ra còn một số hàm thống kê thông dụng như:

- Correlation.
- Covariance.
- Weighted average.
- Median.
- Pth percentile.
- Nth largest.
- Smallest.

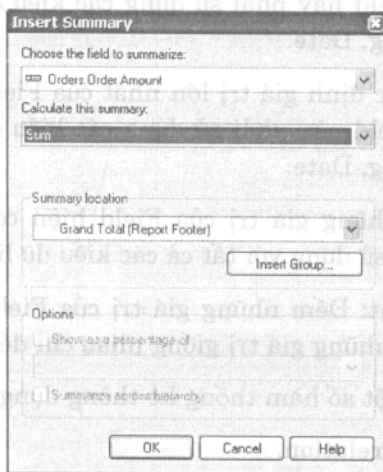
- Mode.
- Nth most frequent.
- Sample variance.
- Sample standard deviation.
- Population variance.

2.2. Thêm Field tổng kết trong báo biểu

Các Field tổng kết đơn giản như tính tổng, trung bình, nhỏ nhất, lớn nhất không yêu cầu thêm các tiêu chí điều kiện nào khác.

Trong ví dụ sau: Chúng ta sẽ thêm một Field tổng kết vào báo biểu tổng kết doanh số hàng bán khiến nó dễ đọc hơn và nêu bật được thông tin chứa trong báo biểu.

Lưu ý: Các báo biểu được tạo trong chương này chứa trong thư mục Project trong đĩa CD ROM.



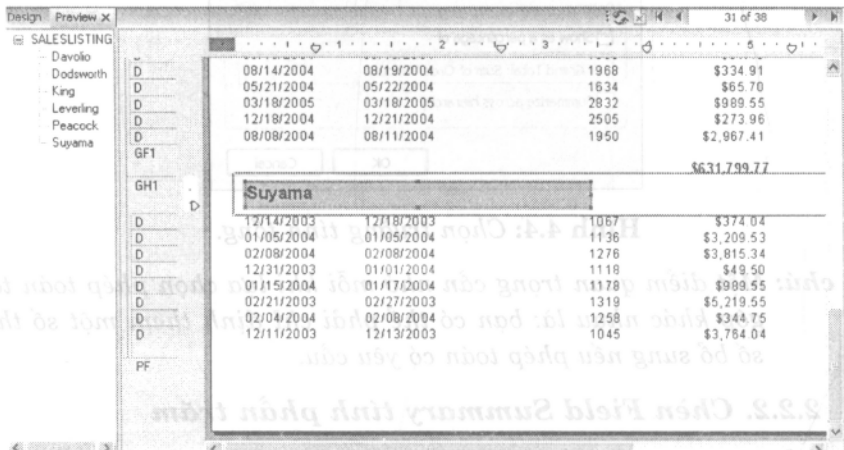
Hình 4.2: Tùy chọn để chèn các trường tổng kết đơn giản trong báo biểu.

Để tạo ra một Field tổng kết đơn giản, hãy thực hiện các bước sau:

1. Mở Crystal Reports và mở báo biểu **Salelisting.PRT** trên CD ROM.
2. Xác định và nhấn chọn vào trường mà bạn muốn tổng kết, trong trường hợp này, ta chọn trường Order Amount.

Ghi chú: Phải chắc chắn rằng trường Order Amount xuất hiện trong phân đoạn Detail của báo biểu. Để cho việc thiết kế được dễ dàng, trước hết bạn hãy chuyển qua chế độ Design.

- Nhấn chọn Insert > Summary, hộp thoại xuất hiện như hình 4.2. Chọn công thức tổng kết mà bạn muốn sử dụng để tính toán trong ô Calculate this Summary. Trong ví dụ này, chọn toán tử tính toán Sum. Chọn nhóm Group #1: Employee.Last Name - A.
- Nhấn chọn OK để chấp nhận thay đổi. Field tổng kết được chèn vào vị trí mà bạn đã chọn như hình 4.3.



Employee	Order Date	Order Amount	Summary
Diavolo	08/14/2004	1968	\$334.91
Dodsworth	05/21/2004	1634	\$65.70
King	03/18/2005	2832	\$989.55
Leverling	12/18/2004	2505	\$273.96
Peacock	08/08/2004	1950	\$2,967.41
Suyama	08/11/2004		\$631,799.77

Hình 4.3: Vị trí trường tính tổng đặt trong báo biểu.

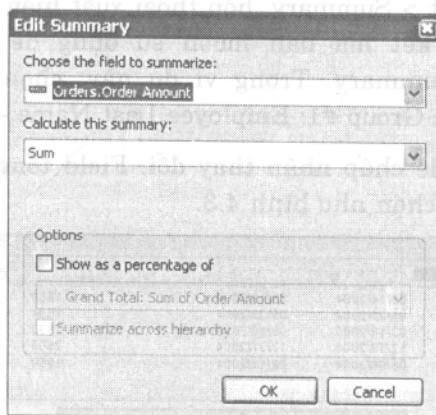
Khi cuộn báo biểu, chú ý bên dưới trường Order Amount bây giờ là một Field Sum xuất hiện tương ứng với mỗi nhóm. Field này tính tổng cộng toàn bộ doanh số bán của một nhân viên. Những Field tính tổng thường sử dụng với nhóm và xuất hiện phía dưới cuối nhóm (phân đoạn Footer Group).

Nếu bạn chọn tùy chọn tính tổng dồn Grand Total, tổng này sẽ luôn luôn xuất hiện trong phần cuối của báo biểu cho biết tổng cộng của toàn báo biểu.

2.2.1. Thay đổi những phép toán tính tổng

Sau khi một Field tính tổng đã được chèn vào trong báo biểu, bạn có thể thay đổi những phép toán tính tổng bằng cách chọn lại Field tính tổng mà bạn muốn thay đổi trên báo biểu, nhấn chọn Field muốn thay đổi, nhấn chọn Edit > Edit Summary.

Trong hộp thoại Edit Summary, nhấn chọn vào ComboBox sẽ hiển thị danh sách các công thức tính tổng sẵn có để thay đổi phép toán tổng gộp, sau đó nhấn chọn OK để chấp nhận thay đổi. Thay đổi sẽ được hiện thị ngay lập tức trong báo biểu.



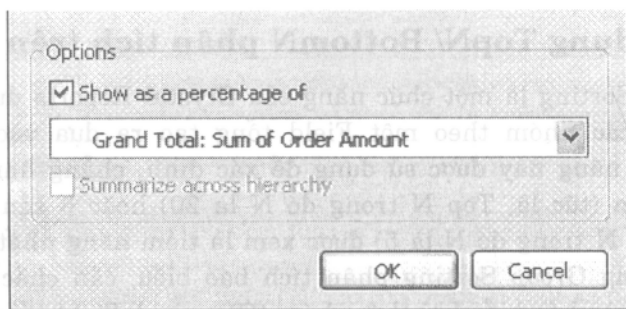
Hình 4.4: Chọn trường tính tổng.

Ghi chú: Một điểm quan trọng cần nhớ mỗi khi lựa chọn phép toán tổng gộp khác nhau là: bạn có thể phải chỉ định thêm một số tham số bổ sung nếu phép toán có yêu cầu.

2.2.2. Chèn Field Summary tính phần trăm

Để chèn Field Summary hiển thị phần trăm trong báo biểu doanh số bán mà chúng ta đang làm việc, hãy thực hiện các bước sau:

1. Mở Crystal Reports và mở báo biểu cần làm việc: báo biểu Salessummary.RPT trên CD ROM.
2. Nhấn chọn Insert > Summary.
3. Trong hộp thoại, lựa chọn kiểu phép toán tổng gộp mà bạn muốn chèn vào từ danh sách Calculate this Summary. Trong ví dụ này, hãy chọn phép toán Sum tính tổng đơn giản.
4. Chọn Field muốn thực hiện tính tổng gộp, và xác định vị trí để đặt Field tổng này. Đối với báo biểu này, chọn trường Order Amount và vị trí tổng gộp được đặt ở nhóm Group #1: Employee. Last Name - A
5. Ở cuối hộp thoại, đánh dấu chọn mục *Show as a Percentage Of*, sử dụng danh sách ComboBox, chọn trường tổng cộng mà bạn muốn tính toán phần trăm như hình 4.5.



Hình 4.5: Trường tổng kết tính theo phần trăm.

6. Nhấn chọn OK để thêm Field tổng kết tính phần trăm vào báo biểu.

Báo biểu bây giờ sẽ hiện tổng doanh số bán của của một nhân viên trong công ty theo phần trăm doanh số của toàn công ty. Bạn có thể định dạng lại Field phần trăm sử dụng một số phương pháp định danh chúng ta đã học trong chương 2.

2.2.3. Chèn các Field tổng dồn Grand Total

Field Grand Total xuất hiện trong phần cuối của báo biểu và được sử dụng để tổng kết nội dung toàn bộ số liệu trong báo biểu. Để chèn Field tổng dồn Grand Total vào báo biểu, bạn thực hiện các bước sau:

1. Mở Cryatal Report và mở báo biểu Salessummary.RPT trên CD ROM.
2. Xác định vị trí và nhấn chọn vào Field muốn tính tổng dồn, trong trường hợp này, chọn trường Order Amount.
3. Nhấn chọn Insert > Summary, sử dụng hộp thoại Insert Summary mà chúng ta vừa xem qua. Chọn kiểu tổng kết và xác định vị trí sẽ đặt giá trị Grand Total. Nhấn chọn OK để chấp nhận thay đổi.

3. Phân tích dữ liệu báo biểu

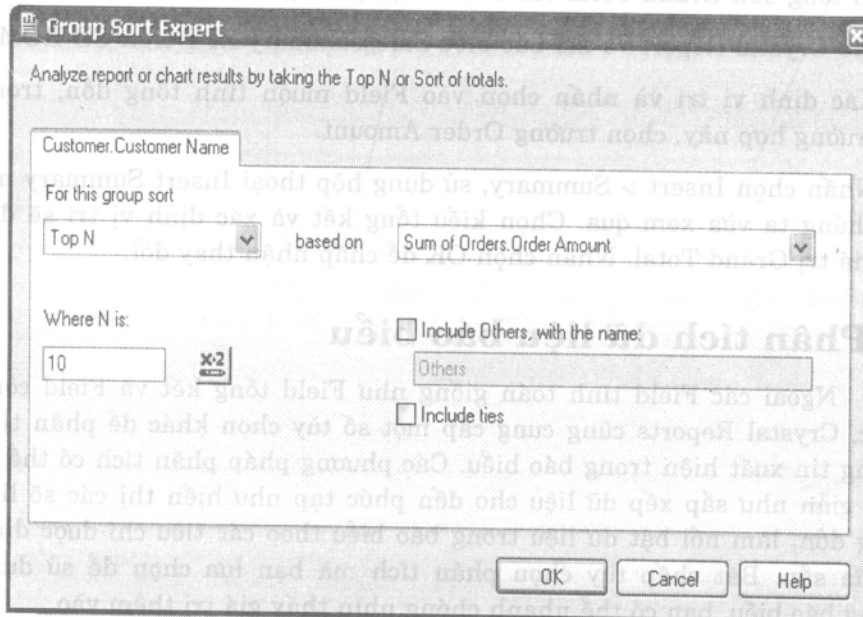
Ngoài các Field tính toán giống như Field tổng kết và Field công thức, Crystal Reports cũng cung cấp một số tùy chọn khác để phân tích thông tin xuất hiện trong báo biểu. Các phương pháp phân tích có thể từ đơn giản như sắp xếp dữ liệu cho đến phức tạp như hiển thị các số liệu tổng dồn, làm nổi bật dữ liệu trong báo biểu theo các tiêu chí được định nghĩa sẵn. Bất chấp tùy chọn phân tích mà bạn lựa chọn để sử dụng trong báo biểu, bạn có thể nhanh chóng nhìn thấy giá trị thêm vào.

3.1. Sử dụng TopN/ BottomN phân tích trên nhóm

Group Sorting là một chức năng của Crystal Reports được sử dụng để sắp xếp các nhóm theo một Field tổng tạo ra dựa vào nhóm đó. Thường chức năng này được sử dụng để xác định, chẳng hạn 20 khách hàng đầu tiên (tức là, Top N trong đó N là 20) hoặc 5 sản phẩm cuối cùng (Bottom N trong đó N là 5) được xem là tiềm năng nhất. Trước khi có thể sử dụng Group Sorting phân tích báo biểu, cần chắc chắn rằng bạn đã chèn vào báo biểu hai thứ: nhóm (Group) và Field tổng kết (Field Summary). Nếu không bạn sẽ không thể sử dụng được chức năng phân tích Group Sorting.

Để thêm Group Sorting vào báo biểu, bạn thực hiện các bước sau:

1. Mở Crystal Reports, mở báo biểu Analysisreport.RPT từ CD ROM.
2. Kiểm tra báo biểu phải có ít nhất một nhóm và Field tổng kết được chèn vào, nếu chưa có bạn hãy chọn một tiêu chí và tự tạo nhóm sau đó tạo Field tổng kết như chúng ta đã học.
3. Nhấn chọn Report > Group Sort Expert, hộp thoại xuất hiện như hình 4.6, bạn có thể chọn Top N hay Bottom N, phần trăm Top N hoặc phần trăm Bottom N. Trong ví dụ này hãy chọn Top N.



Hình 4.6: Tùy chọn cho sắp xếp nhóm.

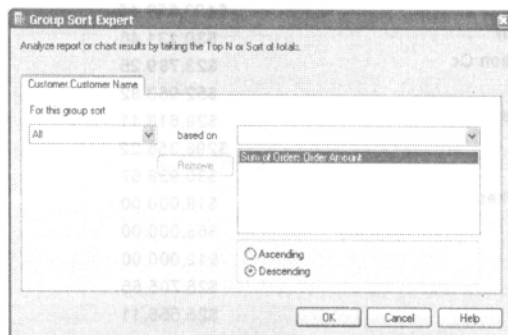
4. Sử dụng danh sách xổ xuống của các Field tổng kết có sẵn, chọn Field sẽ được dùng sắp xếp cho việc phân tích. Chú ý rằng, danh sách này chỉ hiển thị các Field tổng kết mà bạn đã chèn vào báo biểu. Trong báo biểu này, hãy chọn *Sum of Orders.Order Amount*.
5. Tiếp theo, nhập giá trị N. Do chúng ta muốn tạo Top 10 khách hàng trong báo biểu nên bạn hãy nhập N là 10. Tuy nhiên, ngoài các nhóm Top 10, bạn có thể sử dụng ô Checkbox để đưa thêm nhóm khác vào báo biểu.
6. Nhấn chọn OK để chấp nhận thay đổi và áp dụng Group Sorting cho báo biểu.

Bây giờ báo biểu chỉ trình bày Top 10 khách hàng (10 khách hàng đầu tiên). Bạn có thể sử dụng cùng phương pháp để tạo ra một báo biểu chỉ hiển thị Bottom N (N mẫu tin cuối) hoặc theo phần trăm tức là Top 50% khách hàng đầu hay Bottom 30% sản phẩm cuối trong báo biểu.

3.2. Sắp xếp theo tổng nhóm Subtotal hoặc Summary Field

Khi làm việc với nhóm, thỉnh thoảng bạn có thể muốn sắp xếp theo một số Field tổng kết đã chèn vào trong báo biểu. Chức năng này giống tương tự như sắp xếp Group Sorting, chỉ có điều chúng không bị giới hạn trong một nhóm. Sử dụng chức năng Sort All của Crystal Reports, bạn có thể sắp xếp tất cả các nhóm theo một giá trị tổng kết.

Để sắp xếp các nhóm trong báo biểu của bạn theo giá trị của Field tổng kết, kiểm tra báo biểu phải có ít nhất một nhóm và Field tổng kết. Nhấn chọn Report > Group Sorting Expert. Sau đó, trên hộp thoại như hình 4.7, chọn All từ danh sách For This Group Sort.



Hình 4.7: Bạn có thể sắp xếp các nhóm theo một Field tổng kết.

Sử dụng danh sách của các Field tính tổng có sẵn, chọn Field dùng phân tích. Chọn thứ tự sắp xếp (tăng hoặc giảm) và nhấn chọn OK để chấp nhận thay đổi. Các nhóm bây giờ sẽ được sắp xếp bởi Field tổng kết bạn đã chọn.

3.3. Tổng chạy (Running Total)

Tổng chạy hay còn gọi là Running Total cung cấp một cái nhìn phân tích về các giá trị tích lũy trong báo biểu.

Sử dụng tổng chạy trong báo biểu, bạn có thể nhanh chóng cho người sử dụng thấy ngay thông tin tổng tích lũy sau mỗi mẫu tin mà không cần chờ báo biểu chạy xong để xem kết quả tính tổng tại Field tổng kết ở cuối báo biểu hoặc nhóm.

Ví dụ: Giả sử bạn muốn tạo một tổng chạy Running Total trong một cột dọc theo danh sách khách hàng cho biết doanh số bán cộng dồn năm vừa qua của họ, với mỗi mẫu tin, bạn muốn con số bán hàng thể hiện tăng dần theo tổng số đang chạy như hình 4.8:

Running Totals

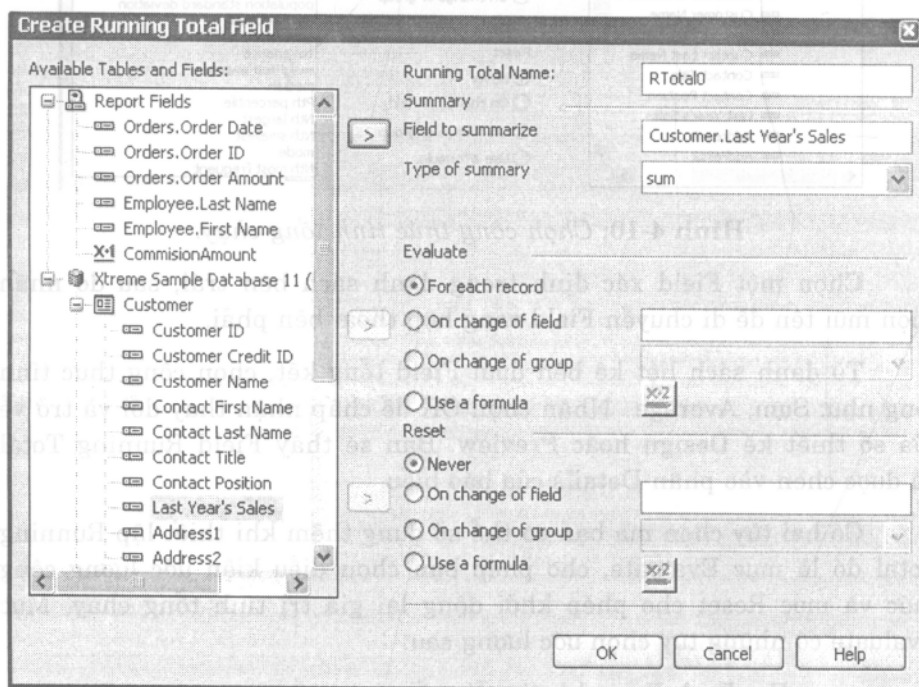
Customer Name	Last Year's Sales	Running Total
City Cyclists	\$20,045.27	\$20,045.27
Pathfinders	\$26,369.63	\$46,414.90
Bike-A-Holics Anonymous	\$4,500.00	\$50,914.90
Psycho-Cycle	\$52,809.11	\$103,724.01
Sporting Wheels Inc.	\$85,642.56	\$189,365.57
Rockshocks for Jocks	\$40,778.52	\$230,145.09
Poser Cycles	\$10,923.00	\$241,068.09
Spokes 'N' Wheels Ltd.	\$25,521.31	\$266,589.40
Trail Blazers Place	\$123,658.46	\$390,247.86
Rowdy Rims Company	\$30,131.46	\$420,379.31
Clean Air Transportation Co	\$23,789.25	\$444,168.56
Hooked on Helmets	\$52,963.82	\$497,132.38
C-Gate Cycle Shoppe	\$29,618.11	\$526,750.49
Alley Cat Cycles	\$298,355.22	\$825,106.71
The Bike Cellar	\$30,938.67	\$856,045.37
Hercules Mountain Bikes	\$18,000.00	\$874,045.37
Whistler Rentals	\$68,000.00	\$942,045.37
Bikes and Trikes	\$12,000.00	\$954,045.37
Changing Gears	\$26,705.65	\$980,751.02
Wheels and Stuff	\$25,556.11	\$1,006,307.13

Hình 4.8: Một ví dụ về báo biểu có tổng số chạy (running total).

Ghi chú: Bạn cũng có thể chèn Field Running Total từ menu Field Object (Insert> Field Object). Xác định phân đoạn để đặt tổng chạy trong Field Explorer, nhấn phím phải và chọn New.

Bước đầu tiên trong việc tạo một Field Running Total là xác định vị trí và chọn Field cần tính tổng chạy.

Nhấn chuột phải vào Field muốn tạo Running Total và nhấn chọn Insert> Running Total, một hộp thoại sẽ xuất hiện như hình 4.9.

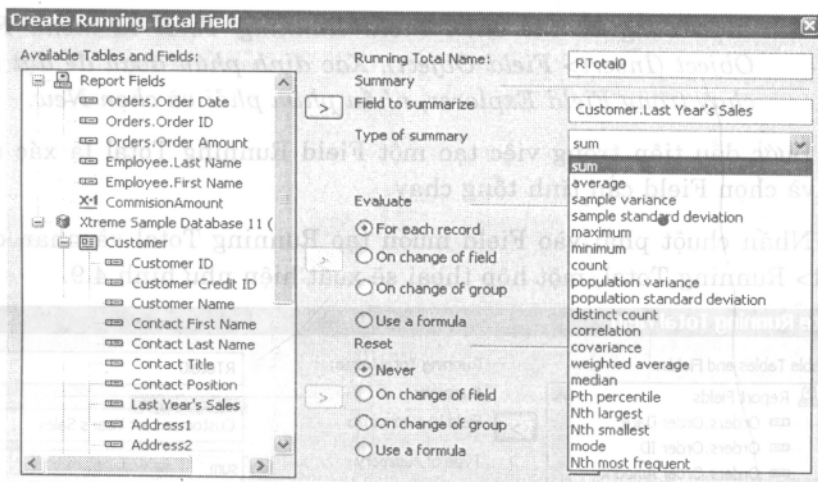


Hình 4.9: Tùy chọn thiết lập tổng chạy.

Bạn phải đặt tên cho Field Running Total trong hộp thoại Running Total Name. Có thể đặt bất kỳ tên nào mà bạn thích, miễn là có ý nghĩa với bạn.

Ghi chú: Crystal Reports sẽ đặt ký hiệu # trước tên của Field Running Total, giúp bạn có thể dễ dàng xác định Field Running Total này khi được chèn trong báo biểu.

Bước tiếp theo trong việc tạo một Field Running Total là chọn Field để tính tổng và công thức tính như hình 4.10.



Hình 4-10: Chọn công thức tính tổng chạy.

Chọn một Field xác định trong danh sách bên trái, sau đó nhấn chọn mũi tên để di chuyển Field sang hộp thoại bên phải.

Từ danh sách liệt kê bên dưới Field tổng kết, chọn công thức tính tổng như: Sum, Average... Nhấn chọn OK để chấp nhận thay đổi và trở về cửa sổ thiết kế Design hoặc Preview. Bạn sẽ thấy Field Running Total đã được chèn vào phần Details của báo biểu.

Có hai tùy chọn mà bạn có thể sử dụng thêm khi thiết lập Running Total đó là mục Evaluate, cho phép bạn chọn điều kiện ước lượng công thức và mục Reset cho phép khởi động lại giá trị tính tổng chạy. Mục Evaluate có những tùy chọn ước lượng sau:

- For Each Record (với mỗi mẫu tin).
- On change of Field (khi giá trị Field thay đổi).
- On change of Group (khi chuyển sang nhóm khác).
- Use a Formula (sử dụng công thức).

Ví dụ, bạn có thể sử dụng tùy chọn này khi tạo Field Running Total hiển thị doanh số bán của tất cả các công ty con trên toàn cầu, bạn chọn Use a Formula và nhập vào điều kiện tính tổng chạy như sau:

[Customer.Country]<> "USA"

Kết quả Field Running Total chỉ được hiển thị khi khách hàng không thuộc USA.

Tương tự như vậy, bạn có thể thiết lập khởi động lại giá trị tính tổng khi đang chạy trong mục Reset theo các sự kiện sau:

- Never (chạy liên tục không thay đổi giá trị).
- On change of Field (quay về 0 khi giá trị Field thay đổi).
- On change of Group (quay về 0 khi nhóm thay đổi).
- Use a Formula (sử dụng công thức tùy biến).

Ví dụ, bạn có thể thiết lập một Field Running Total sao cho mỗi khi thay đổi giá trị của Field Country trong danh sách thì giá trị tổng chạy sẽ được tính lại. Bằng việc sử dụng tùy chọn Evaluation và Reset, bạn có thể tính được mọi trường hợp tùy biến cho tổng chạy Running Total theo yêu cầu của báo biểu.

3.3.1. Điểm sáng (highlight) báo biểu

Bạn có thể điểm sáng một vùng hay giá trị nào đó trong báo biểu theo một điều kiện chỉ định thông qua hộp thoại Highlighting Expert như: thay phông chữ, màu nền và những đường viền khi có một điều kiện thỏa mãn.

Để sử dụng Highlighting Expert trong báo biểu, bạn chọn Field có giá trị muốn làm nổi bật. Nhấn chuột phải vào Field, chọn Highlighting Expert từ menu. Hộp thoại Highlighting Expert xuất hiện như hình 4.11.



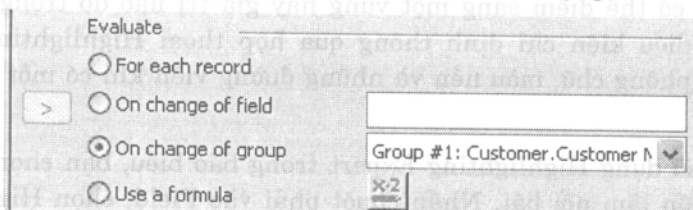
Hình 4-11: Hộp thoại Highlighting Expert.

Bạn nhập điều kiện bên phải của hộp thoại, sau đó chỉ rõ những tùy chọn định dạng mà bạn muốn sử dụng khi điều kiện thỏa mãn. Nếu muốn nhập vào nhiều điều kiện, trong danh sách nhấn chọn New Item. Để thay đổi thứ tự của điều kiện, bạn nhấn chọn vào mũi tên lên và xuống. Nhấn chọn OK chấp nhận thay đổi. Các Field mà bạn chọn định dạng theo điều kiện trước đây sẽ có tác động ngay lập tức.

3.3.2. Chèn Field Running Total cho một nhóm

Ngoài việc sử dụng Field Running Total cho một danh sách xuyên suốt báo biểu, bạn có thể đặt Running Total cho một nhóm. Để chèn Field Running Total cho nhóm, bạn chọn Field muốn tính tổng chạy, nhấn chuột phải vào Field và nhấn chọn Insert > Running Total.

Do chúng ta cần tạo Field Running Total cho nhóm nên tại mục Evaluate, bạn chọn For Each Record và dưới mục Reset bạn chọn On change of Group, sau đó chọn nhóm mà bạn muốn sử dụng như hình 4.12.



Hình 4.12: Tạo Running Total cho nhóm.

Nhấn chọn OK để chấp nhận thay đổi. Field Running Total cho nhóm sẽ xuất hiện trong phần chân của nhóm (Group Footer).

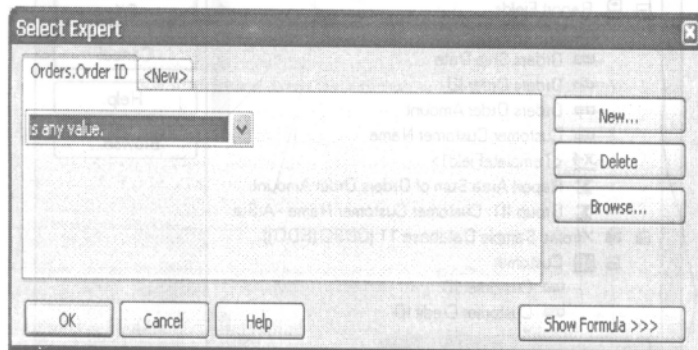
4. SỬ DỤNG RECORD SELECTION ĐỂ LỌC MẪU TIN TRONG BÁO BIỂU

Khi tạo một báo biểu, có thể bạn không muốn sử dụng tất cả các mẫu tin lưu trong cơ sở dữ liệu mà chỉ sử dụng một tập hợp các mẫu tin có trạng thái đặc biệt, vùng, phạm vi ngày tháng nào đó. Quá trình tuyển chọn để bỏ bớt số mẫu tin trả về được gọi là chọn mẫu tin hay Record Selection. Quá trình chọn lọc mẫu tin sử dụng ngôn ngữ công thức của Crystal Reports để tạo ra biểu thức logic chỉ hiển thị những mẫu tin thỏa mãn theo yêu cầu lọc. Công thức chọn lọc mẫu tin có thể sử dụng đoạn mã như sau:

```
{customer.country}="USA"
```

Tất cả mẫu tin được đọc từ cơ sở dữ liệu và ước lượng bởi công thức này. Chẳng hạn khi báo biểu được in, bạn sẽ thấy chỉ có những khách hàng của Mỹ hay USA.

Nếu việc tự viết công thức làm bạn khó chịu thì bạn có thể dùng chức năng Select Expert như hình 4.13.



Hình 4.13: Select Expert trong Crystal Reports.

Select Expert là một hộp thoại chuyên dụng được thiết kế giúp bạn tạo ra tiêu chuẩn chọn lọc nhanh chóng mà không cần viết công thức.

Select Expert có thể áp dụng tiêu thức lọc cho một hoặc đồng thời nhiều Field.

4.1. Áp dụng lọc mẫu tin đơn giản dựa trên một Field

Lọc các mẫu tin dựa trên giá trị của một Field là cách dễ nhất để hình dung cách hoạt động của Select Expert. Trong ví dụ trước, chúng ta đã tạo một công thức chọn lọc mẫu tin dựa trên giá trị của một Field chỉ lọc ra những khách hàng nào thuộc quốc gia United States:

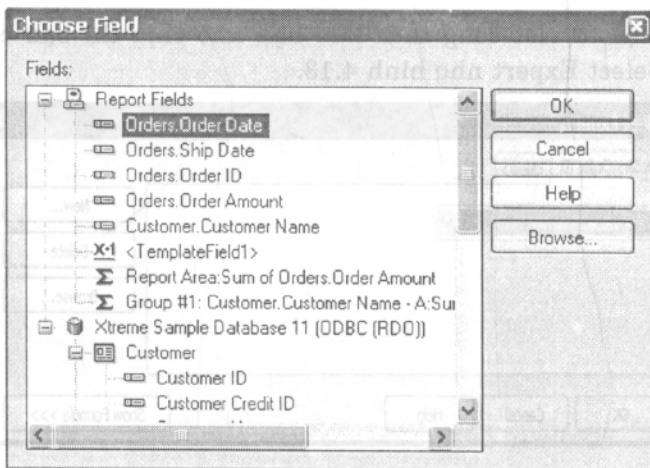
```
{Customer.Country} = "USA"
```

Chúng ta có thể tạo trực tiếp công thức này như đã làm, nhưng ở đây ta sẽ sử dụng Select Expert để thực hiện công việc này.

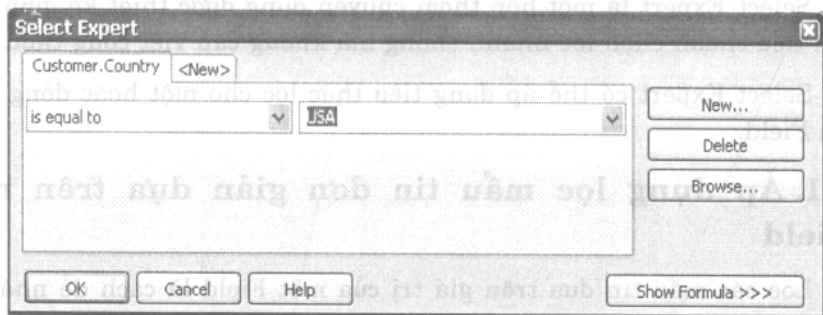
Để bắt đầu, bạn nhấn chọn Report > Select Expert. Một danh sách các Field hiển thị để bạn sử dụng chọn lọc mẫu tin (xem hình 4.14).

Chọn Field dùng làm tiêu chí lọc mẫu tin và nhấn OK. Sử dụng hộp thoại như hình 4.15, bạn chọn phép toán lọc mẫu tin và nhập vào dữ liệu lọc tương ứng. Ví dụ, bạn chọn phép toán bằng nhau (Equal To) và

sau đó nhập giá trị “USA”. Nhấn chọn OK để chấp nhận thay đổi. Báo biểu bây giờ lọc và giữ lại chỉ những khách hàng trong phạm vi quốc gia United States.



Hình 4.14: Chọn Field để làm tiêu chí lọc mẫu tin.



Hình 4.15: Chọn phép toán và nhập vào điều kiện lọc.

Ghi chú: Chúng ta cũng có thể sử dụng danh sách xổ xuống để chọn những giá trị trực tiếp từ cơ sở dữ liệu, nhưng đặc tính này chỉ hiển thị và cho phép chọn giá trị của 200 mẫu tin đầu tiên từ bảng dữ liệu, và giá trị mà bạn đang tìm kiếm có thể không có trong danh sách.

Bảng 4.1: Mô tả các phép toán.

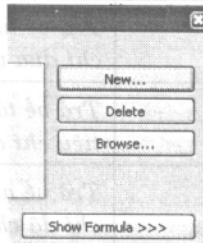
Phép toán	Mô tả
Is any value	Trả về tất cả các mẫu tin có trong cơ sở dữ liệu, bất chấp giá trị nào.

Is equal to	Trả về những mẫu tin theo một tiêu chí đưa vào.
Is not equal to	Trả về tất cả những mẫu tin khác với tiêu chí đưa vào.
Is one of	Trả về những mẫu tin theo nhiều tiêu chí đã chọn (cho phép chọn nhiều giá trị của một Field nào đó).
Is not one of	Trả về tất cả những mẫu tin khác với nhiều tiêu chí đã chọn (cho phép chọn nhiều giá trị của một Field nào đó).
Is less than	Trả về bất kỳ mẫu tin nào ít hơn tiêu chí đưa vào.
Is not less than or equal to	Trả về bất kỳ mẫu tin nào ít hơn hoặc bằng tiêu chí đưa vào.
Is greater than	Trả về bất kỳ mẫu tin nào lớn hơn tiêu chí đưa vào.
Is greater than or equal to	Trả về bất kỳ mẫu tin nào lớn hơn hoặc bằng tiêu chí đưa vào.
Is not between	Trả về những mẫu tin nào ngoài phạm vi hai tiêu chí đưa vào.
Is between	Trả về tất cả những mẫu tin nằm giữa phạm vi của hai tiêu chí đưa vào.
Formula	Sử dụng công thức chọn lọc mẫu tin không có trong Select Expert.

4.2. Áp dụng chọn lọc mẫu tin cho nhiều Field

Select Expert cũng có thể sử dụng để áp tiêu chí chọn lọc cho nhiều Field. Để thiết lập tiêu chí lọc cho nhiều Field, bạn thực hiện cùng quá trình như cho một Field: Nhấn chọn Report > Select Expert. Sau đó nhấn chọn vào nút New như hình 4.16.

Sau khi nhấn nút New, một danh sách các Field sẽ xuất hiện. Từ danh sách này bạn có thể chọn các Field sử dụng cho mục đích chọn lọc mẫu tin, một Tab thứ hai xuất hiện, bạn có thể chỉ rõ phép toán, giá trị lọc v.v...



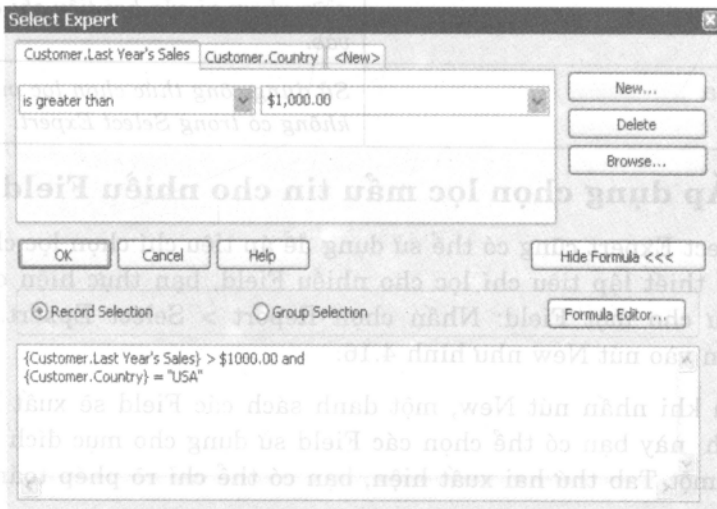
Hình 4.16: Bạn nhấn chọn nút *New* để thêm tiêu chí cho nhiều *Field*.

Ghi chú: Mỗi khi bạn sử dụng nhiều *Field*, *Crystal Reports* sẽ liên kết giữa hai tiêu chí bằng phép toán *AND* (nghĩa là, điều kiện *Field 1* và điều kiện *Field 2* phải đúng thì *Crystal Reports* mới sử dụng mẫu tin).

Để xóa tiêu chí, nhấn chọn vào Tab có tên *Field* bạn muốn xóa và sử dụng nút *Delete* để loại bỏ Tab đó.

4.3. Sử dụng phát biểu *OR* trong công thức chọn lọc mẫu tin

Khi sử dụng *Select Expert* và chọn lọc mẫu tin trên nhiều *Field*, *Crystal Reports* mặc định xét mối quan hệ giữa hai *Field* theo mệnh đề *AND*, xem hình 4.17. Trong ví dụ này, các mẫu tin được lọc hiển thị trong báo biểu khi *Field Country* là *USA* và *Field Last Year's Sales* lớn hơn \$10.000.



Hình 4.17: *Crystal Reports* đặt phép toán *AND* mặc định giữa hai *Field*.

Nếu muốn sử dụng phép toán OR, bạn có thể nhấn chọn vào nút Show Formula bên góc phải dưới và sửa bằng tay công thức chọn lọc mẫu tin.

Ghi chú: Một khi đã sửa công thức chọn lọc mẫu tin bằng tay, không thể sử dụng Select Expert để sửa công thức được nữa. Một thông báo có thể xuất hiện như sau: "Composite Expression: Please use formula editor to do editing" ("Biểu thức phức hợp: Xin sử dụng trình soạn thảo cho việc sửa công thức"). Thông báo này cho biết Select Expert không phân tích được công thức do bạn cung cấp và bạn phải hoàn toàn "tự biên tự diễn" lấy công thức của mình.

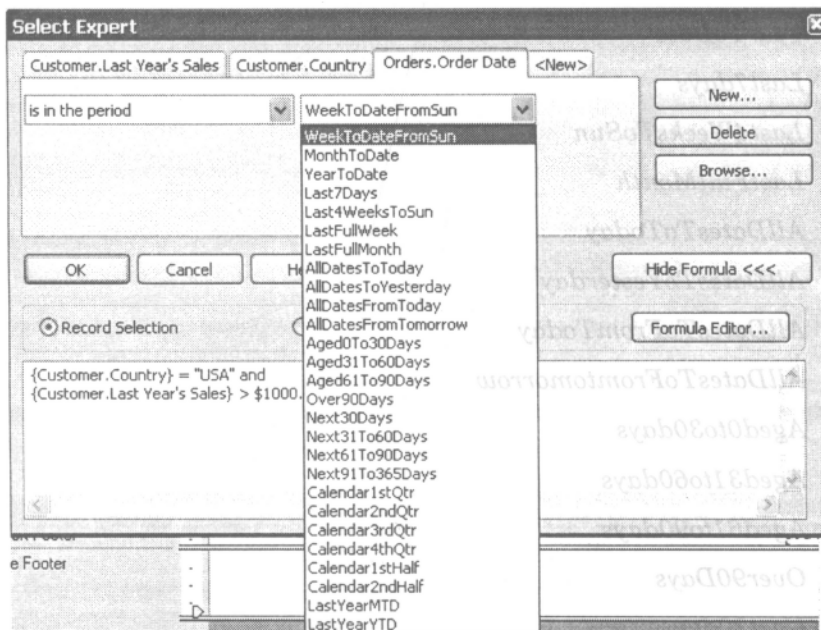
4.4. Áp dụng chọn lọc mẫu tin dựa vào những Field Date (Ngày tháng)

Với những Field ngày tháng, chúng ta có thêm các phép toán lọc mẫu tin khác xác định khoảng cách thời gian (In the Period). Dưới đây là một số mẫu thời gian định sẵn:

- *WeekToDateFromSun*
- *MonthToDate*
- *YearToDate*
- *Last7days*
- *Last4WeeksToSun*
- *LastFullMonth*
- *AllDatesToToday*
- *AllDatesToYesterday*
- *AllDatesToFromToday*
- *AllDatesToFromtomorrow*
- *Aged0to30days*
- *Aged31to60days*
- *Aged61to90days*
- *Over90Days*
- *Next30days*
- *Next31to60days*

- *Next61to90days*
- *Calender1stQtr*
- *Calender2ndQtr*
- *Calender3rdQtr*
- *Calender4thQtr*
- *Calender1stHalf*
- *Calender2ndHalf*
- *LastYearMtd*
- *LastYearYtd*

Để sử dụng những khoảng thời gian này với các Field trong báo biểu, bạn nhấn chọn Report> Select Expert. Từ hộp thoại Choose Fiels, chọn Field ngày tháng mà bạn muốn sử dụng chọn lọc trong báo biểu, nhấn chọn OK, hộp thoại Select Expert xuất hiện. Sử dụng danh sách xổ xuống đầu tiên, xem hình 4.18, chọn một trong hai phép toán *Is in the Period* hoặc *Not in the Period*. Chọn trong danh sách liệt kê thứ hai những khoảng thời gian có sẵn.

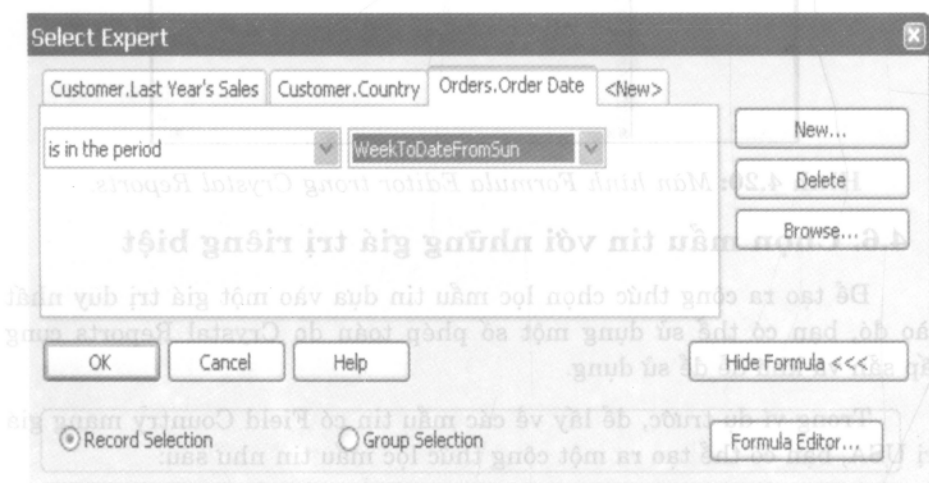


Hình 4.18: Chỉ định lọc mẫu tin theo thời gian.

Nhấn OK chấp nhận áp dụng các tiêu chí lọc và quay trở lại báo biểu, bây giờ dữ liệu sẽ hiển thị theo đúng với điều kiện thời gian bạn chỉ định.

4.5. Viết công thức chọn lọc mẫu tin

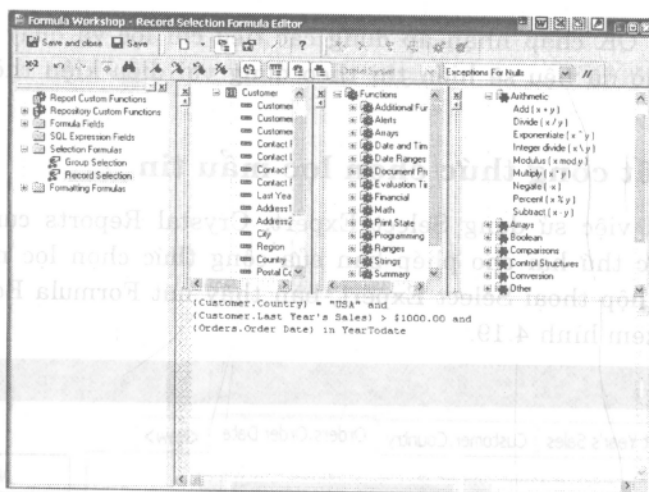
Ngoài việc sử dụng Select Expert, Crystal Reports cung cấp một phương thức thứ hai cho phép bạn sửa công thức chọn lọc mẫu tin thủ công. Trên hộp thoại Select Expert, bạn thấy nút Formula Editor... ở góc phải dưới, xem hình 4.19.



Hình 4.19: Nhấn chọn vào nút *Formula Editor...* để hiệu chỉnh công thức.

Bằng cách xem công thức mà Crystal Reports tạo ra, bạn có thể biết cách mà công thức làm việc. Để viết các công thức lọc của riêng mình, bạn nhấn chọn Report> Selection Formula. Hộp thoại Record Selection Formula Editor sẽ xuất hiện.

Sử dụng Formula Editor, xem hình 4.20, bạn có thể tạo một công thức chọn lọc mẫu tin mà kết quả trả về là một giá trị đại số Bool (ước lượng giá trị đúng hoặc sai). Sử dụng nút X+2 trong thanh công cụ để kiểm tra cú pháp công thức lọc mẫu tin. Khi bạn đã hoàn thành công việc với công thức chọn lọc, nhấn nút *Save and Close* để lưu công thức và đóng màn hình Formula Editor. Báo biểu bây giờ sẽ thể hiện các mẫu tin theo tiêu chí lọc mà bạn đã tạo.



Hình 4.20: Màn hình Formula Editor trong Crystal Reports.

4.6. Chọn mẫu tin với những giá trị riêng biệt

Để tạo ra công thức chọn lọc mẫu tin dựa vào một giá trị duy nhất nào đó, bạn có thể sử dụng một số phép toán do Crystal Reports cung cấp sẵn và khá dễ để sử dụng.

Trong ví dụ trước, để lấy về các mẫu tin có Field Country mang giá trị USA, bạn có thể tạo ra một công thức lọc mẫu tin như sau:

```
{Customer.country}="USA"
```

Và ngoài phép toán bằng, bạn cũng có thể sử dụng các phép toán khác. Ví dụ, để hiển thị tất cả các mẫu tin có doanh số bán lớn hơn 10000\$ bạn tạo công thức như sau:

```
{Customer.last year is sales}=10000
```

Chú ý, với các Field chuỗi, chúng ta đặt điều kiện vào trong dấu ngoặc kép. Đối với những Field số, ta không cần đặt điều kiện trong dấu ngoặc kép.

Khi làm việc với Field dữ liệu kiểu ngày tháng hoặc Field dữ liệu kiểu thời gian, chúng ta cũng có thể sử dụng cùng các phép toán này. Ví dụ, để trả về tất cả mẫu tin đơn hàng trong cơ sở dữ liệu có ngày mua trước 01/08/2003, công thức tính như sau:

```
{Purchases.PurchaseDate} < date (2003,08,01)
```

Và nếu Field *PurchaseDate* là Field dữ liệu kiểu thời gian đầy đủ, bạn xây dựng công thức như sau:

```
{ Purchases.PurchaseDate }  
    <DateTime(2003,08,01,00,00,00)
```

Chú ý, khi sử dụng Field ngày tháng trong một công thức lọc mẫu tin, bạn phải chuyển đổi giá trị sang dạng ngày tháng bằng một trong hai hàm *Date* hoặc *DateTime* do ngôn ngữ Crystal Reports cung cấp.

4.7. Chọn mẫu tin dựa vào nhiều giá trị

Nếu muốn lọc mẫu tin dựa vào nhiều giá trị, bạn có thể sử dụng kết hợp nhiều phương pháp khác nhau, tùy vào các giá trị được sắp xếp như thế nào. Chẳng hạn, nếu muốn sử dụng những giá trị được sắp xếp theo một khoảng (như từ 1-30, từ A-K, v.v...), bạn có thể sử dụng phép toán *In* và so sánh giá trị của Field trong cơ sở dữ liệu với khoảng giá trị bằng từ khóa *To*. Ví dụ:

```
{Customer.Last Year's Sales} In 1000 To 3000
```

Công thức tương tự khi sử dụng chuỗi:

```
{Customer.Initial} in "A" to "K"
```

Với những giá trị nằm ngoài một phạm vi, bạn sử dụng từ khóa *In* và dấu ngoặc vuông, các giá trị so sánh đặt trong cặp ngoặc vuông phân cách nhau bằng dấu phẩy. Ví dụ, nếu muốn chỉ hiển thị các mẫu tin của khách hàng ở các quốc gia: United States, Canada và Mexico, bạn tạo công thức như sau:

```
{Customer.Country} in ["Canada", "Mexico", "USA"]
```

Hay với giá trị số:

```
{Customer.Last Year's Sales} in [1000,2000,3000]
```

Và bạn cũng có thể kết hợp hai phương pháp so sánh như sau:

```
{Customer.Last Year's Sales} in [  
    1000 to 2000,3000,4000]
```

Ghi chú: Bạn cũng có thể sử dụng các giá trị khoảng thời gian như *MonthToDate*, *YearToDate* trong công thức lọc mẫu tin, ví dụ:

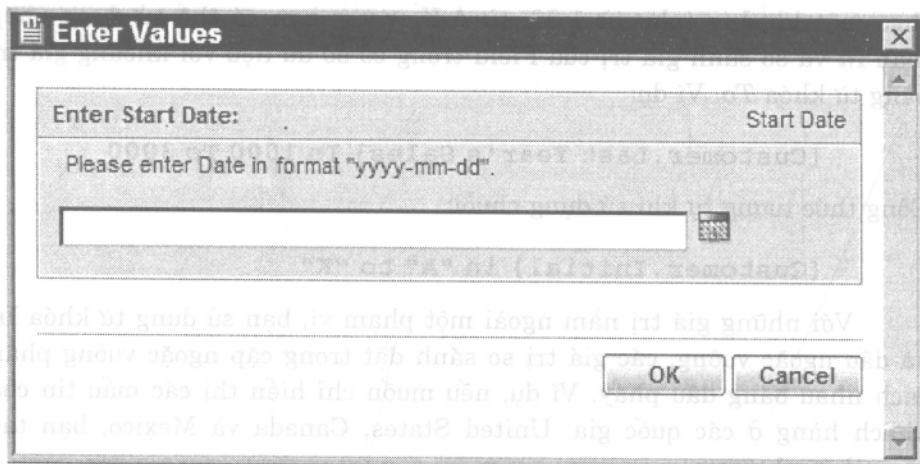
```
{Order.Order date} in MonthToDate)
```

5. SỬ DỤNG THAM SỐ TRONG BÁO BIỂU

Một trong những mục đích mà báo biểu thiết kế tốt là thực hiện chuyển giao thông tin tới nhiều người, không phải sử dụng nhiều báo biểu để trình bày thông tin trùng lặp. Một phương pháp mà chúng ta có thể sử dụng để tạo ra những báo biểu thay đổi theo tình huống đó là sử dụng các Field tham số (Parameter Field).

Trong phần sau, chúng ta sẽ học về Field số, cách tạo và sử dụng chúng trong thiết kế báo biểu.

Các Field tham số được sử dụng trong báo biểu thông qua một hộp thoại yêu cầu người dùng nhập thông tin tiêu chí (hay giá trị tham số) vào (hình 4.21).



Hình 4.21: Nhập tham số vào hộp thoại.

Field tham số cũng giống như mọi Field khác sử dụng trong báo biểu. Chúng có thể hiển thị, sử dụng trong điều kiện lọc mẫu tin. Trong các phiên bản trước của Crystal, Field tham số sử dụng rất hạn chế.

Field Tham số đã được cải thiện đáng kể trong Crystal Reports XI và giờ đây, bạn có thể sử dụng tham số lọc từ nguồn dữ liệu hay lọc các tham số theo từng cấp (cascade). Ví dụ, nếu bạn có hai tham số cho người dùng nhập vào là Field Country và Field State trong báo biểu, khi người dùng chọn một quốc gia (Country) từ danh sách thì các bang hay thành phố tương ứng (State) của quốc gia đó cũng sẽ hiển thị tương ứng.

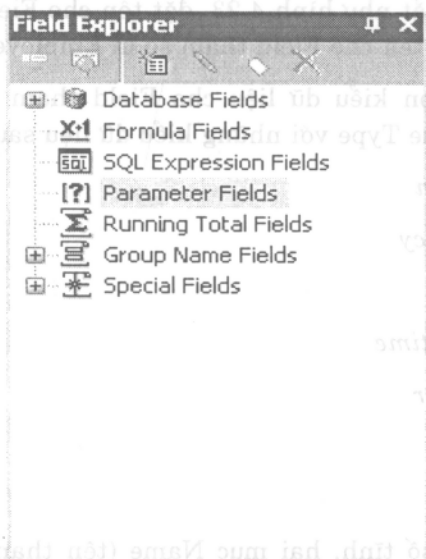
Với những đặc tính mới này, Crystal Reports XI có hai kiểu tham số khác nhau:

- *Static Parameter (tham số tĩnh)*: Tương tự tham số trong các phiên bản trước của Crystal, tham số tĩnh có thể hiển thị một danh sách những giá trị đưa vào trước hoặc trích rút từ nguồn dữ liệu. Những giá trị tĩnh này được lưu cố định trong báo biểu.
- *Dynamic Parameter (tham số động)*: Tham số này có thể kết hợp với nguồn cơ sở dữ liệu, do vậy khi bạn thêm một giá trị mới vào cơ sở dữ liệu thì danh sách các giá trị tham số cũng sẽ thay đổi. Tham số động cũng có thể lọc theo cấp (cascade) tức là giá trị tham số này sẽ dùng để lọc danh sách các giá trị trong tham số khác.

Trong những phần sau, chúng ta sẽ học về cả hai kiểu tham số cũng như cách sử dụng chúng để lọc ra những mẫu tin kết quả cần tìm trong báo biểu.

5.1. Làm việc với những Field tham số tĩnh

Để tạo ra tham số tĩnh trong báo biểu, bạn nhấn chọn View> Field Explorer. Màn hình Field Explorer của Crystal Reports xuất hiện như hình 4.22.



Hình 4.22: Màn hình Field Explorer của Crystal Reports.

Create New Parameter

Name: Type:

List of Values: ☒ Static ☐ Dynamic

Value Field: Description Field:

Actions ▾

Value	Description
Click here to add item	

Options:

Option	Setting
Prompt Text	Enter Employee:
Prompt With Description Only	False
Default Value	
Allow custom values	True
Allow multiple values	False
Allow discrete values	True

Hình 4.23: Tạo Field tham số tĩnh.

Nhấn chuột phải vào Parameter Fields trong cửa sổ Field Explorer, chọn New từ menu tắt như hình 4.23, đặt tên cho Field tham số; trong ví dụ này chúng ta đặt tên cho Field tham số là Employee.

Tiếp theo, chọn kiểu dữ liệu cho Field tham số bằng cách nhấn chọn danh sách Value Type với những kiểu dữ liệu sau:

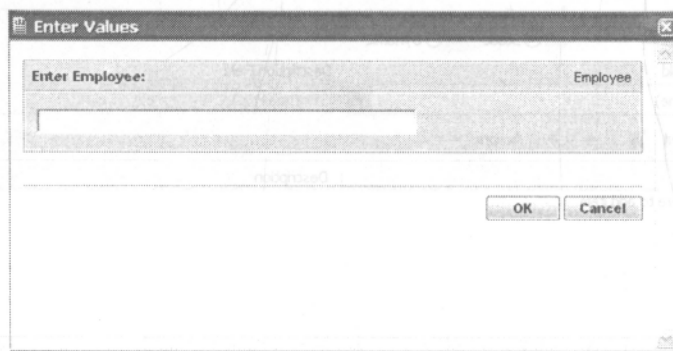
- Boolean
- Currency
- Date
- Date - time
- Number
- String
- Time

Để tạo tham số tĩnh, hai mục Name (tên tham số) và kiểu Value Type bắt buộc phải có, bạn nhấn chọn OK để chấp nhận thay đổi và quay

trở lại báo biểu. Field tham số sẽ xuất hiện bên trong cửa sổ Field Explorer và sẵn sàng để sử dụng trong báo biểu.

5.2. Chèn một Field tham số vào báo biểu

Để chèn một Field tham số mà bạn đã tạo vào báo biểu, đơn giản bạn có thể kéo nó từ cửa sổ Field Explorer đặt lên báo biểu trong chế độ Design. Khi chuyển sang chế độ Preview, một hộp thoại sẽ xuất hiện yêu cầu bạn nhập giá trị tham số như hình 4.24.



Hình 4.24: Hộp thoại nhập giá trị tham số.

Sau khi bạn nhập vào một giá trị cho Field tham số, giá trị đó sẽ được sử dụng trong báo biểu cho đến khi bạn làm tươi lại chế độ Preview thì tham số tĩnh sẽ được yêu cầu nhập vào giá trị mới.

5.3. Thiết lập giá trị tham số mặc định

Để giúp người dùng giảm thời gian nhập liệu, Crystal cho phép đặt các giá trị mặc định sẽ được chọn.

Danh sách những giá trị tham số mặc định có thể được đọc từ cơ sở dữ liệu hoặc nhập vào bằng tay. Một khái niệm quan trọng khi làm việc với những giá trị tham số mặc định đó là những giá trị này cần phải tạo ra trước trong chế độ thiết kế.

Nếu muốn nhập vào một danh sách của những giá trị rời rạc, bạn có thể nhấn chọn nút Insert trong hộp thoại tham số và sau đó nhập vào các giá trị cho tham số trong khung lưới bên dưới. Nếu muốn đọc những giá trị rời rạc từ cơ sở dữ liệu, bạn thực hiện các bước như sau:

1. Mở Crystal Reports và mở báo biểu Commission.RPT từ đĩa CD ROM.

2. Nhấn chọn View> Field Explorer để bung ra một danh sách các Field trong báo biểu.
3. Nhấn chuột phải trên mục Parameter Fields, chọn New từ menu tắt. Hộp thoại Create New Parameter xuất hiện như hình 4.25.

Create New Parameter

Name: Type:

List of Values: ☒ Static ☐ Dynamic

Value Field: Description Field:

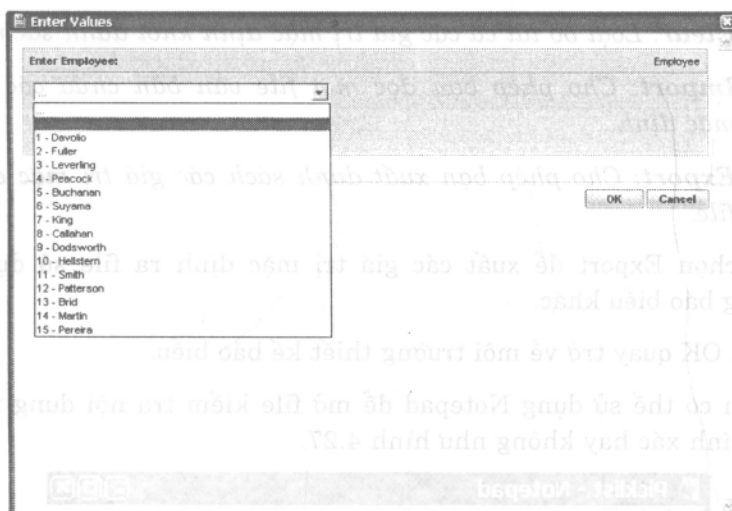
Value	Description
Click here to add item	

Options:

Option	Setting
Prompt Text	Enter Employee:
Prompt With Description Only	False
Default Value	False
Allow custom values	True
Allow multiple values	False
Allow discrete values	True

Hình 4.25: Màn hình tạo tham số.

4. Trong ô Name, bạn nhập vào tên tham số là Employee và chọn kiểu Number từ danh sách xổ xuống bên phải.
5. Tiếp theo, trong danh sách Value bạn nhấn chọn Field mang tên Employee ID.
6. Trong danh sách Description, bạn nhấn chọn Field mang tên Last Name.
7. Nhấn chọn vào nút Actions và chọn giá trị Append All Database.
8. Nhấn chọn OK quay trở lại báo biểu.
9. Kéo Field tham số Employee vào trong Page Header của báo biểu, sau đó chuyển báo biểu sang chế độ Preview để xem kết quả (hình 4-26).



Hình 4-26: Kết quả danh sách chọn các giá trị cho Field tham số.

5.4. Sắp xếp các giá trị Field tham số mặc định

Để xem danh sách các giá trị mặc định dễ dàng hơn, chúng ta có thể sắp xếp theo nội dung hiển thị.

Khi tạo danh sách các giá trị mặc định, bạn có thể nhấn chọn cột Value hoặc Description của khung lưới. Các giá trị mặc định trong khung lưới sẽ được sắp xếp tương ứng.

5.5. Nhập/Xuất các giá trị trong danh sách Field tham số

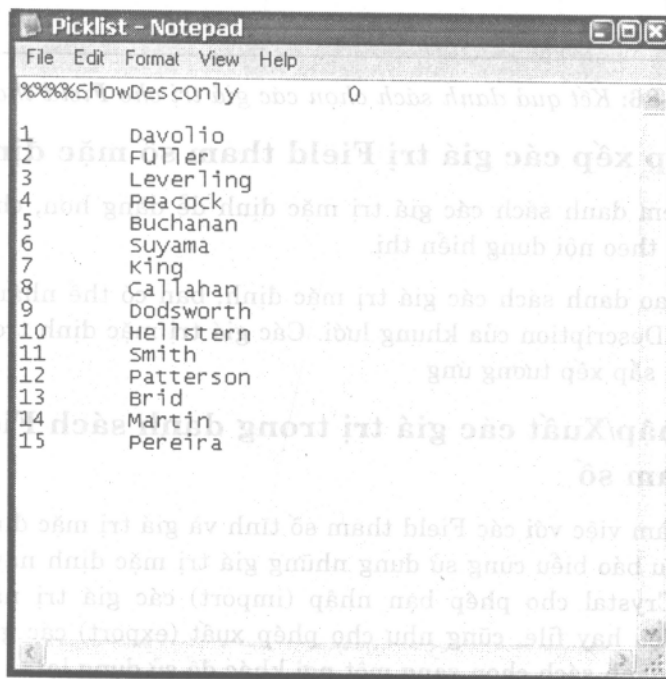
Khi làm việc với các Field tham số tĩnh và giá trị mặc định, bạn có thể có nhiều báo biểu cùng sử dụng những giá trị mặc định này. Để khỏi nhập lại, Crystal cho phép bạn nhập (import) các giá trị này từ một nguồn dữ liệu hay file, cũng như cho phép xuất (export) các giá trị mặc định trong danh sách chọn sang một nơi khác để sử dụng lại.

Để nhập/xuất các giá trị trong danh sách Field tham số, bạn thực hiện các bước sau:

1. Nhấn chọn View > Field Explorer.
2. Nhấn chuột phải vào Field tham số Employee và chọn Edit từ menu, hộp thoại Edit Parameter xuất hiện. Ở dưới nút Actions, một danh sách xổ xuống cung cấp ba tùy chọn để làm việc với các giá trị mặc định:

- **Clear:** Loại bỏ tất cả các giá trị mặc định khỏi danh sách chọn.
 - **Import:** Cho phép bạn đọc một file văn bản chứa các giá trị mặc định.
 - **Export:** Cho phép bạn xuất danh sách các giá trị mặc định ra file.
3. Bạn chọn Export để xuất các giá trị mặc định ra file sử dụng cho những báo biểu khác.
 4. Nhấn OK quay trở về môi trường thiết kế báo biểu.

Bạn có thể sử dụng Notepad để mở file kiểm tra nội dung xuất dữ liệu có chính xác hay không như hình 4.27.

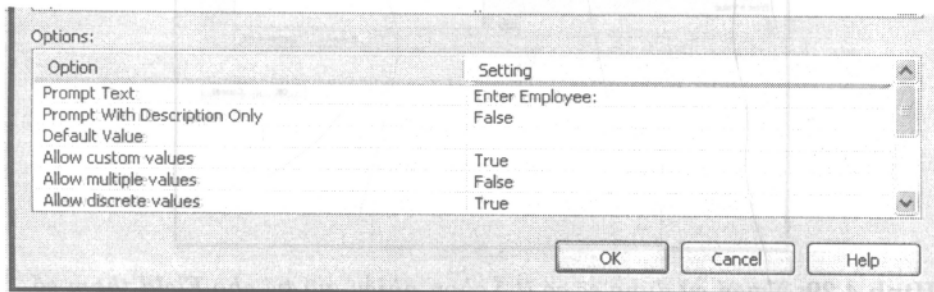


Hình 4.27: Xuất/nhập các giá trị tham số.

Ghi chú: Bạn cũng có thể sử dụng nút Import Pick List để đọc vào một danh sách các giá trị cho tham số từ một file văn bản có cùng định dạng như file xuất.

5.6. Tùy biến chuỗi nhắc tham số

Khi người dùng được yêu cầu nhập giá trị tham số trong hộp thoại, bạn có thể tùy biến nội dung hiển thị của các nhãn mô tả ý nghĩa tham số như hình 4.28. Một số tùy chọn khác bạn có thể thay đổi là:



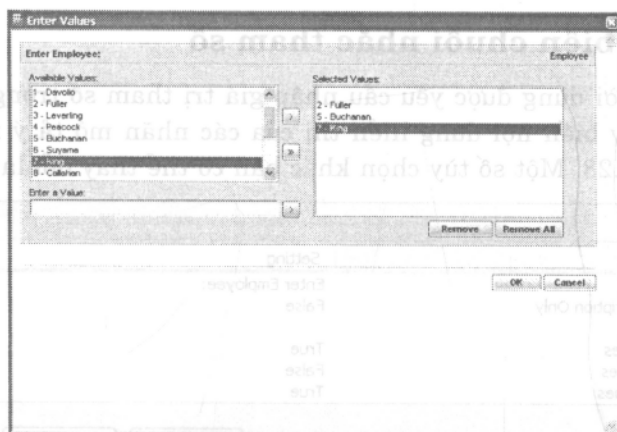
Hình 4.28: Tùy biến nhãn mô tả tham số.

- **Prompt Text:** Mô tả ý nghĩa của Field tham số để người sử dụng hiểu và nhập giá trị vào.
- **Prompt With Description Only:** Mô tả nội dung mà không cần hiển thị giá trị.
- **Default Value:** Đặt giá trị mặc định cho một Field.
- **Allow Custom Value:** Cho phép người sử dụng nhập giá trị tùy biến vào tham số.

5.7. Tạo Field tham số nhiều giá trị

Ngoài những giá trị đơn và dãy giá trị, Field tham số có thể cho phép người dùng nhập vào nhiều giá trị. Ví dụ, bạn có thể tạo ra một Field tham số là Country, sau đó để cho người dùng nhập vào danh sách các nước mà họ muốn chạy báo biểu. Họ có thể nhập một nước, hai nước thậm chí là 30 nước nếu muốn.

Để tạo ra một Field tham số cho phép nhận nhiều giá trị, bạn tạo Field tham số như cách thông thường, sau đó thay đổi thuộc tính tùy chọn Allow là True. Nhấn OK chấp nhận thay đổi và trở lại chế độ thiết kế. Khi người sử dụng chạy báo biểu Field tham số sẽ được yêu cầu nhập. Một hộp thoại như hình 4.29 sẽ xuất hiện cho phép bạn chọn một danh sách các giá trị.



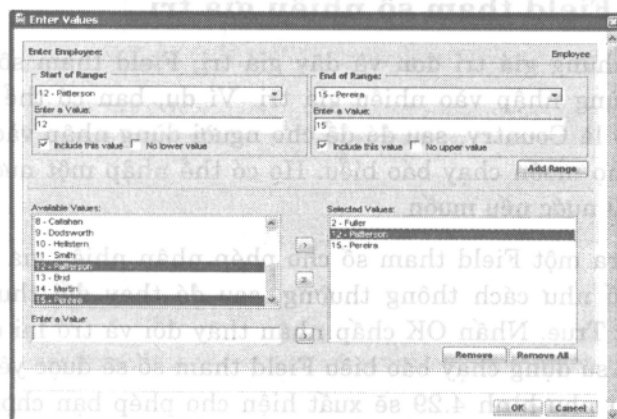
Hình 4.29: Người sử dụng sẽ có thể chọn nhiều giá trị cho Field tham số.

5.8. Giới hạn các giá trị nhập của tham số

Khi sử dụng Field tham số, chúng ta có thể buộc người dùng chỉ được nhập vào giá trị trong phạm vi nào đó.

Chèn một Field tham số mới như cách bình thường, nhưng bây giờ bạn sử dụng danh sách xổ xuống Options để thay đổi tùy chọn Allow range thành giá trị True.

Khi tham số được chèn vào trong báo biểu hoặc được sử dụng với bộ lọc mẫu tin, Crystal sẽ yêu cầu bạn nhập vào giá trị trong phạm vi chỉ định như hình 4.30.



Hình 4.30: Bạn có thể định nghĩa một tham số để chấp nhận các giá trị được chọn nằm trong phạm vi nào đó.

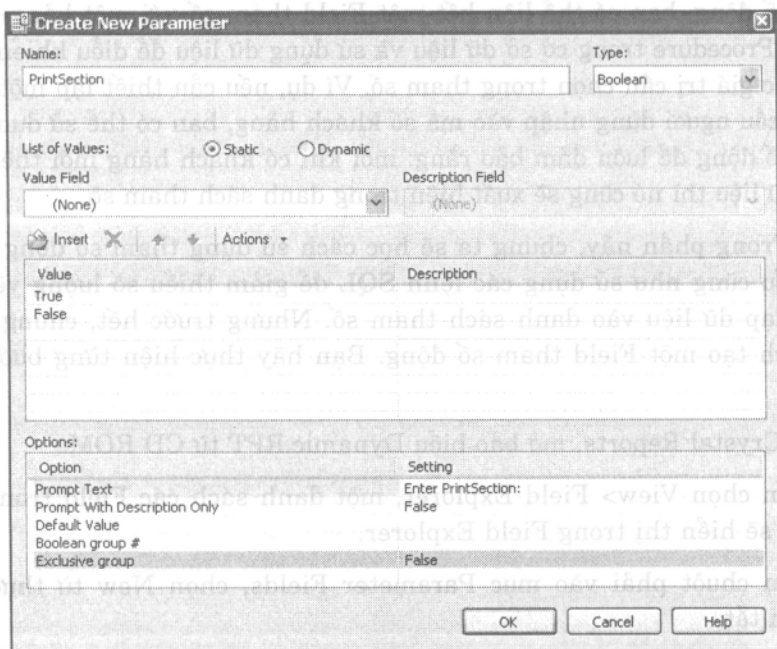
5.9. Sử dụng tham số tĩnh điều khiển các tính năng của báo biểu

Những Field tham số kiểu Boolean có thể sử dụng để yêu cầu người dùng trả lời là đúng (True) hoặc sai (False) và do đó có thể sử dụng để điều khiển định dạng, tổng kết của báo biểu.

Ví dụ, bạn có thể tạo ra một báo biểu yêu cầu người sử dụng trả lời câu hỏi “Hiển thị số âm hay không?” và sau đó sử dụng giá trị nhập vào của tham số trong công thức lọc mẫu tin để tuyển chọn ra các giá trị âm hoặc dương như yêu cầu.

Tạo Field tham số Boolean cũng giống như tạo các Field tham số khác. Bạn nhấn chọn View> Field Explorer. Cửa sổ Field Explorer hiển thị. Nhấn chuột phải vào mục Parameter Fields trên cửa sổ Field Explorer, chọn New từ menu tắt. Hộp thoại Create New Parameter xuất hiện, nhập vào tên Field tham số và chọn kiểu là Boolean.

Tiếp theo, sử dụng tùy chọn Prompt Text, nhập vào bất kỳ văn bản nào mà bạn muốn xuất hiện để nhắc nhở người sử dụng nhập thông tin (ví dụ bạn nhập Enter PrintSection) (hình 4.31).



Create New Parameter

Name: Type:

List of Values: ☒ Static ☐ Dynamic

Value Field: Description Field:

Actions ▾

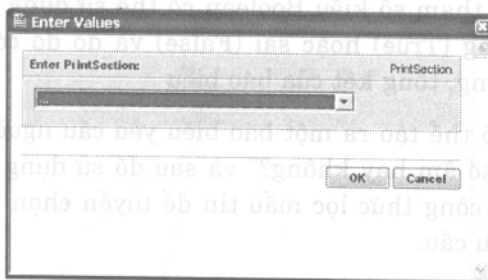
Value	Description
True	
False	

Options:

Option	Setting
Prompt Text	Enter PrintSection:
Prompt With Description Only	False
Default Value	
Boolean group #	
Exclusive group	False

Hình 4.31: Field tham số kiểu Boolean.

Khi người sử dụng được nhắc nhập vào một Field tham số giá trị đúng sai kiểu Boolean, một hộp thoại như hình 4.32 sẽ xuất hiện, bạn có thể chọn để đưa vào giá trị True hoặc False.



Hình 4.32: Chọn giá trị đại số Boolean.

Nếu không muốn chọn giá trị True hoặc False, bạn có thể đưa vào các giá trị đúng sai khác tương đương True/False như Yes/No, On/Off thông qua sử dụng các tùy chọn ở góc dưới của trang.

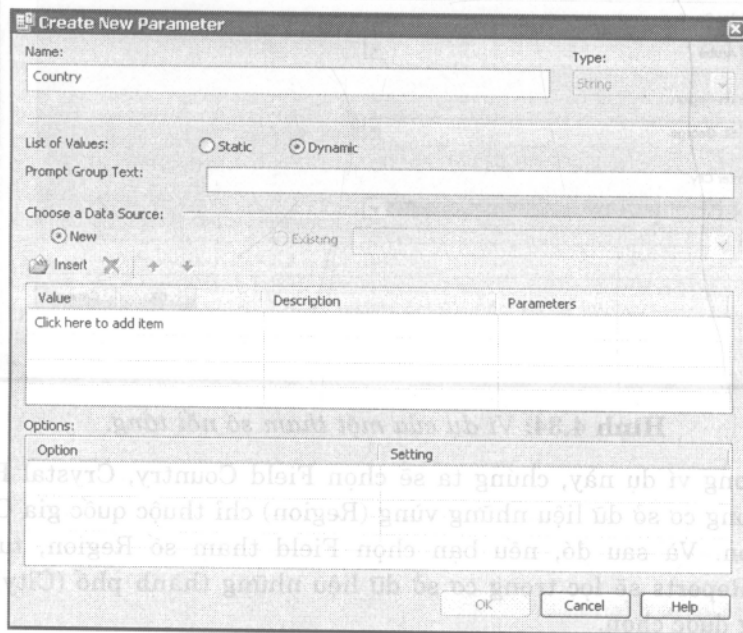
5.10. Làm việc với tham số động

Tham số động là một tính năng mới của Crystal Reports XI. Sử dụng tham số động, bạn có thể liên kết một Field tham số với một bảng, View, Stored Procedure trong cơ sở dữ liệu và sử dụng dữ liệu để điều khiển danh sách các giá trị cần chọn trong tham số. Ví dụ, nếu cần thiết lập một tham số yêu cầu người dùng nhập vào mã số khách hàng, bạn có thể sử dụng một tham số động để luôn đảm bảo rằng: mỗi khi có khách hàng mới thêm vào cơ sở dữ liệu thì nó cũng sẽ xuất hiện trong danh sách tham số.

Trong phần này, chúng ta sẽ học cách sử dụng tham số động trong báo biểu cũng như sử dụng các lệnh SQL để giảm thiểu số lượng yêu cầu xử lý nạp dữ liệu vào danh sách tham số. Nhưng trước hết, chúng ta sẽ học cách tạo một Field tham số động. Bạn hãy thực hiện từng bước như sau:

1. Mở Crystal Reports, mở báo biểu Dynamic.RPT từ CD ROM.
2. Nhấn chọn View> Field Explorer, một danh sách các Field trong báo biểu sẽ hiển thị trong Field Explorer.
3. Nhấn chuột phải vào mục Parameter Fields, chọn New từ thực đơn phím tắt.
4. Nhập tên cho tham số (trong trường hợp này là Country), chọn kiểu của tham số là String.

5. Nhấn chọn vào tùy chọn Dynamic, hộp thoại Create New Parameter sẽ thay đổi như hình 4.33.



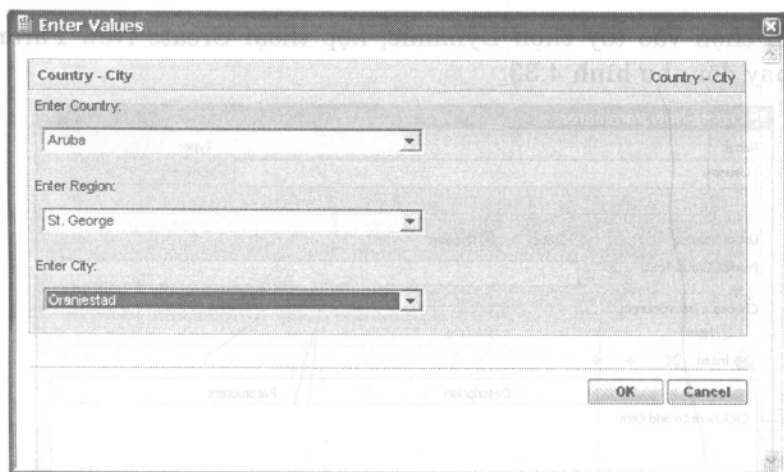
Hình 4.33: Tùy chọn tham số động.

6. Nhấn chọn mục *Click here to add item* để chọn Field, trong ví dụ này bạn chọn Field là Country.
7. Tiếp theo, nhập bất kỳ câu thông báo mà bạn muốn xuất hiện để nhắc nhở người sử dụng chọn giá trị cho Field tham số. Trong ví dụ này, nhập câu thông báo “Please select a country from the list”.
8. Nhấn chọn OK để quay trở về báo biểu.
9. Kéo Field tham số vào báo biểu. Một hộp thoại sẽ xuất hiện nhắc nhở bạn chọn giá trị từ danh sách xổ xuống khi muốn xem báo biểu.

Danh sách giá trị của tham số là danh sách động. Nếu bạn sửa đổi dữ liệu trong CSDL thì nội dung danh sách cũng sẽ thay đổi theo tương ứng.

5.11. Sử dụng tham số động với giá trị lọc nổi tầng

Chỉ đến Crystal Reports XI mới có khái niệm của tham số nổi tầng. Nó có ý nghĩa là bạn sử dụng giá trị tham số này để lọc ra danh sách các giá trị tham số khác. Hình 4.34 là một ví dụ minh họa Field tham số nổi tầng.

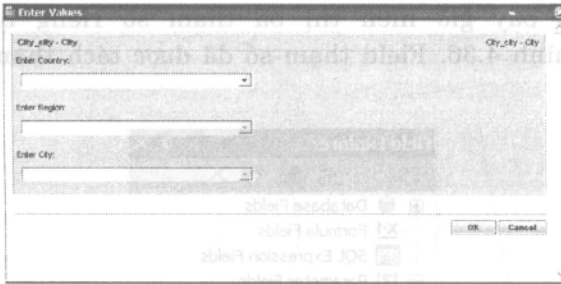


Hình 4.34: Ví dụ của một tham số nổi tầng.

Trong ví dụ này, chúng ta sẽ chọn Field Country, Crystal Reports sẽ lọc trong cơ sở dữ liệu những vùng (Region) chỉ thuộc quốc gia Country được chọn. Và sau đó, nếu bạn chọn Field tham số Region, tương tự Crystal Reports sẽ lọc trong cơ sở dữ liệu những thành phố (City) tương ứng vùng được chọn.

Bạn có thể dễ dàng áp dụng kỹ thuật này cho các báo biểu của mình. Trong ví dụ sau, chúng ta sẽ sử dụng kỹ thuật này để tạo ra tham số nổi tầng được dùng trong báo biểu, bạn thực hiện theo các bước sau:

1. Mở Crystal Reports và mở báo biểu Cascading.PRT từ CD ROM.
2. Nhấn chọn View> Field Explorer để mở danh sách các Field.
3. Nhấn chuột phải vào mục Parameter Fields và chọn New từ menu tắt.
4. Nhập tên cho tham số (trong trường hợp này, nhập tên là City) và chọn kiểu String.
5. Nhấn chọn vào nút Dynamic.
6. Nhấn vào mục *Click here to add item* để chọn Field, trong ví dụ này bạn chọn Field là Country.
7. Trên dòng tiếp theo, chọn Field Region từ danh sách xổ xuống.
8. Và cuối cùng, trên dòng thứ ba, chọn Field City từ danh sách xổ xuống. Khi xem báo biểu ở chế độ Preview, hộp thoại Field tham số bây giờ sẽ trông giống như hình 4.35.



Hình 4.35: Đặt tham số động.

9. Nhấn chọn OK để quay trở lại báo biểu.
10. Kéo Field tham số vừa tạo vào trong báo biểu.

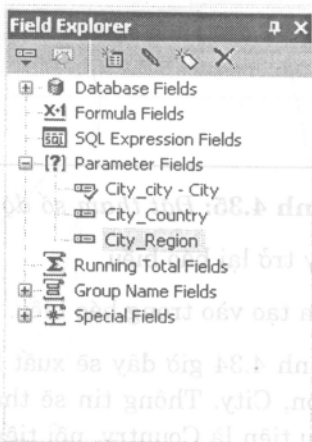
Hộp thoại trong hình 4.34 giờ đây sẽ xuất hiện và yêu cầu bạn chọn các Field: Country, Region, City. Thông tin sẽ thay đổi theo từng tầng khi bạn chọn giá trị, tầng đầu tiên là Country, nối tiếp là Region và sau cùng là City.

5.12. Tách tham số nối tầng thành những Field riêng biệt

Sẽ có lúc bạn muốn tách một tham số nối tầng ra thành nhiều Field. Trong ví dụ trước, bạn có thể tách thành nhiều Field tham số cho phép người sử dụng chọn các giá trị tham số cho các Field Country, Region, City. Để tách Field tham số nối tầng thành những Field riêng biệt, bạn thực hiện từng bước sau:

1. Mở Crystal Reports, sau đó mở báo biểu nối tầng mà bạn đang làm việc ở phần trước.
2. Nhấn chọn View> Field Explorer để mở danh sách các Field.
3. Mở rộng mục Parameter Fields, nhấn chuột phải vào tham số City và chọn Edit từ menu tắt.
4. Từ hộp thoại Edit Parameter, trong khung lưới đang hiển thị giá trị mô tả và Field tham số, bạn kích vào để tạo tham số trên Field Customer.Country.
5. Lặp lại các bước trên cho Field Region.
6. Nhấn chọn OK để quay lại báo biểu.

Báo biểu bây giờ hiển thị ba tham số riêng biệt trên Field Explorer như hình 4.36. Field tham số đã được tách thành những Field riêng biệt.



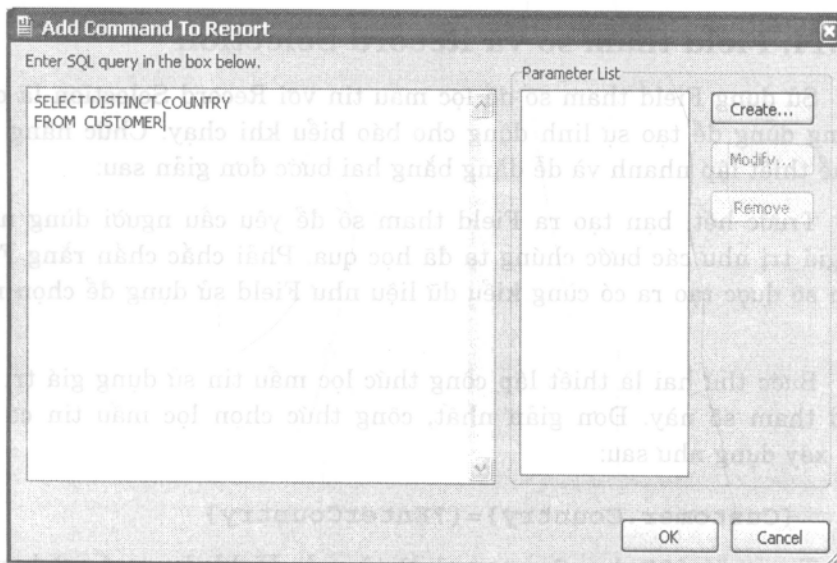
Hình 4.36: Field tham số được tách thành những Field riêng biệt.

5.13. Sử dụng đối tượng lệnh (Command) cho tham số động

Một trong những cách giảm tải việc xử lý dữ liệu của tham số động là sử dụng câu lệnh SQL. Câu lệnh SQL dùng cho tham số động được sử dụng để lấy về chỉ những dữ liệu cần thiết. Ví dụ, nếu chúng ta dựa vào một tham số động yêu cầu người sử dụng nhập vào giá trị Country, chúng ta sẽ dùng lệnh SQL để lọc ra những danh sách quốc gia cần thiết.

Bảng dữ liệu khách hàng có hàng nghìn mẫu tin về khách hàng thuộc nhiều quốc gia Country khác nhau, có khách hàng thuộc 2 quốc gia. Chúng ta sẽ dùng câu lệnh SQL để lọc ra danh sách các quốc gia không trùng lặp trong bảng khách hàng Customer.

1. Mở Crystal Reports và mở báo biểu Parameter.PRT từ CD ROM.
2. Nhấn chọn Database>Database Expert.
3. Nhấn chọn Create New Connection> ODBC RDO mở danh sách ODBC dữ liệu nguồn sẵn.
4. Chọn Xtreme Sample Database 11 và nhấn chọn Finish.
5. Tiếp theo, bên nhánh ODBC (RDO)/ Xtreme Sample Database, bạn nhấp đôi chuột vào mục chọn Add Command. Một hộp thoại xuất hiện như hình 4.37.



Hình 4.37: Hộp thoại Add Command.

6. Nhập câu lệnh SQL sau vào hộp thoại:
SELECT DISTINCT COUNTRY FROM CUSTOMER
7. Nhấn chọn OK hai lần để đóng Database Expert và tiếp theo nhấn chọn OK hai lần nữa để quay trở lại báo biểu.
8. Tiếp theo, nhấn chọn View> Field Explorer để mở danh sách các Field.
9. Nhấn chuột phải vào mục Parameter Fields và chọn New từ menu tắt.
10. Nhập tên cho tham số (trong trường hợp là "Country") và chọn kiểu string.
11. Kích nút tùy chọn Dynamic.
12. Nhấn chọn mục *Click here to add item*, chọn Field Country từ danh sách phát sinh do câu lệnh SQL của bạn.
13. Nhấn chọn OK để quay về báo biểu.
14. Kéo Field tham số vừa tạo vào báo biểu.

Bây giờ nếu bạn nhấn chọn Preview xem báo biểu, một hộp thoại nhắc nhở chọn tham số xuất hiện, câu lệnh SQL sẽ được sử dụng để lọc bỏ những dữ liệu trùng lặp trong danh sách quốc gia.

5.14. Field tham số và Record Selection

Sử dụng Field tham số để lọc mẫu tin với Record Selection là cách thường dùng để tạo sự linh động cho báo biểu khi chạy. Chức năng này có thể thiết lập nhanh và dễ dàng bằng hai bước đơn giản sau:

Trước hết, bạn tạo ra Field tham số để yêu cầu người dùng nhập vào giá trị như các bước chúng ta đã học qua. Phải chắc chắn rằng Field tham số được tạo ra có cùng kiểu dữ liệu như Field sử dụng để chọn mẫu tin.

Bước thứ hai là thiết lập công thức lọc mẫu tin sử dụng giá trị của Field tham số này. Đơn giản nhất, công thức chọn lọc mẫu tin có thể được xây dựng như sau:

`{Customer.Country}={?EnterCountry}`

Trong đó `{?EnterCountry}` là tên của Field tham số mà bạn đã tạo.

Ghi chú: Nếu không thích tự viết công thức, bạn có thể sử dụng *Select Expert*.

Trong ví dụ này, bất cứ khi nào chạy báo biểu, một hộp thoại sẽ hiển thị và yêu cầu bạn nhập vào tên quốc gia, như "USA". Sau khi nhận được giá trị, công thức lọc mẫu tin sẽ thay thế giá trị này ở vị trí tham số.

Bạn cũng có thể tạo ra hai tham số trong báo biểu dùng làm điều kiện lọc, ví dụ như để lọc các mẫu tin có trường ngày tháng, chúng ta dùng Field tham số là *StartDate* và *EndDate*, công thức lọc mẫu tin sẽ được xây dựng như sau:

`{Orders.Order Date} in {?StartDate} to {?EndDate}`

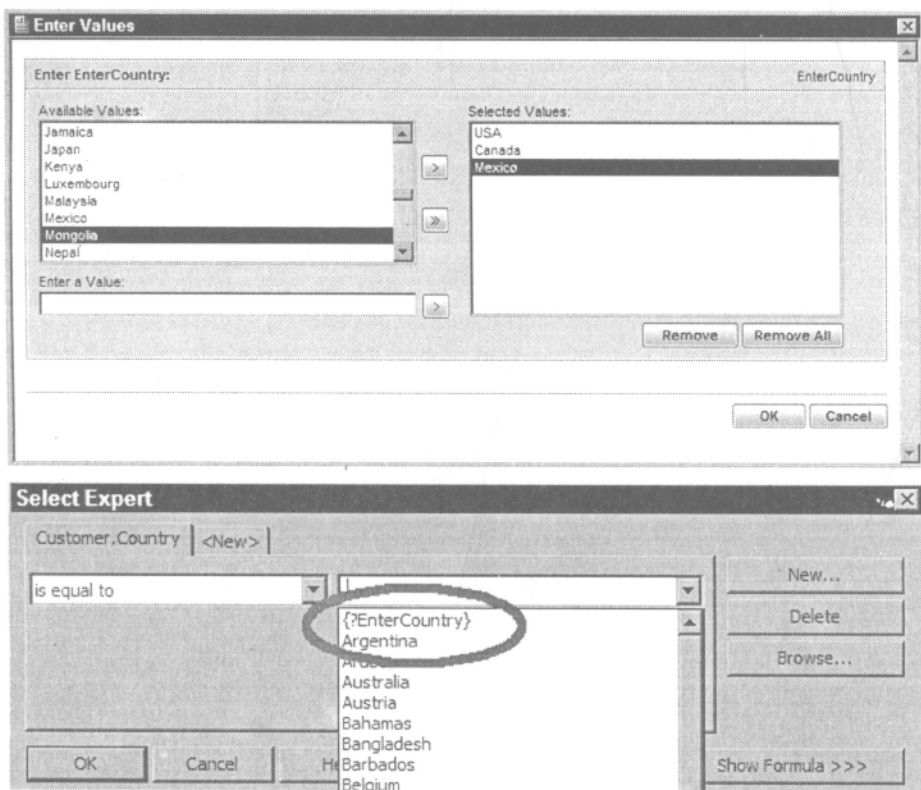
5.15. Sử dụng một Field tham số lọc mẫu tin đơn giản

Field tham số cũng có thể được sử dụng trong một số việc khác, nhưng cách sử dụng phổ biến nhất của Field tham số là dùng lọc các mẫu tin trong báo biểu theo một giá trị và kết quả hiển thị của báo biểu sẽ thu hẹp theo giá trị nhập vào.

Để sử dụng một Field tham số lọc mẫu tin, bạn chỉ định Field cơ sở dữ liệu mà mình muốn sử dụng làm tiêu chí lọc, chú ý kiểu và chiều dài của Field. Sau đó bạn tạo một Field tham số có cùng kiểu với Field

trong cơ sở dữ liệu. Để sử dụng Field này lọc mẫu tin, nhấn chọn *Report > Select Expert*. Một danh sách các Field sẽ xuất hiện, chọn Field cần lọc. Hộp thoại *Select Expert* hiển thị cho phép bạn chọn các phép toán dùng lọc mẫu tin (Equal to, Is Not equal to, Is one of,...).

Khi đã chọn xong phép toán lọc từ danh sách thứ nhất, một danh sách thứ hai sẽ hiển thị như hình 4.38, cho phép bạn chọn hoặc nhập vào một giá trị sẽ sử dụng trong công thức lọc mẫu tin. Sử dụng danh sách xổ xuống, xác định phép toán và chọn Field tham số mà bạn vừa tạo.



Hình 4.38: Chọn Field tham số cho công thức lọc.

Nhấn chọn OK để chấp nhận thay đổi và quay về báo biểu. Mỗi khi bạn chuyển sang Preview hoặc làm tươi (refresh) lại báo biểu, một hộp thoại sẽ hiển thị, nhắc bạn nhập Field tham số mà bạn vừa tạo.

6. TÓM TẮT

Phân tích dữ liệu báo biểu là cách tốt nhất khiến cho báo biểu của bạn có ý nghĩa và thông tin thể hiện có giá trị hơn. Trong chương này, chúng ta đã xem qua hai kỹ thuật khác nhau dùng phân tích dữ liệu báo biểu đó là kỹ thuật sử dụng dạng báo biểu tổng kết, tổng chạy và kỹ thuật lọc báo biểu sử dụng Field tham số. Trong chương tiếp theo, chúng ta sẽ học cách xây dựng hàm, công thức toán học và các biểu thức tính toán trong báo biểu.

Chương 5:

BIỂU THỨC, CÔNG THỨC VÀ HÀM

- ❖ Tổng quan.
- ❖ Giới thiệu công thức.
- ❖ Cách hoạt động của công thức.
- ❖ Tạo các công thức đơn giản.
- ❖ Sử dụng các hàm trong Crystal Reports.
- ❖ Thêm các công thức Logic.
- ❖ Tìm hiểu công thức tính thời gian (Evaluation times).
- ❖ Bẫy lỗi công thức.
- ❖ Cách hoạt động của các hàm tùy biến.
- ❖ Câu lệnh SQL và biểu thức.
- ❖ Tóm tắt.

1. TỔNG QUAN

Crystal Reports cung cấp một môi trường linh hoạt để thực hiện các tính toán phức tạp trong báo biểu, bạn có thể sử dụng ngôn ngữ công thức, câu lệnh SQL và các biểu thức có thể xử lý ngay trên máy chủ cơ sở dữ liệu nhằm tận dụng sức mạnh tính toán của máy chủ.

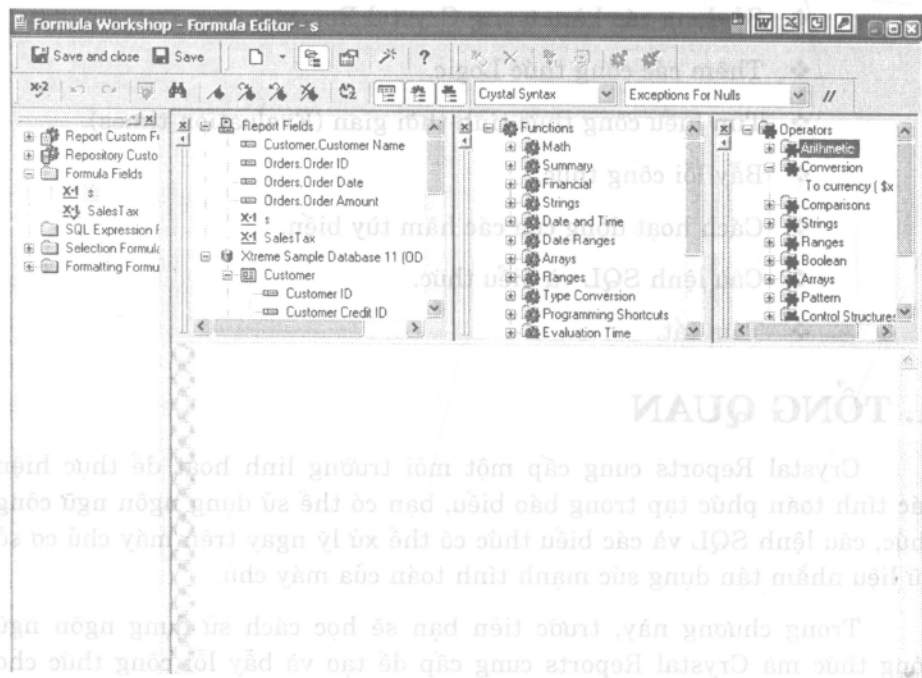
Trong chương này, trước tiên bạn sẽ học cách sử dụng ngôn ngữ công thức mà Crystal Reports cung cấp để tạo và bẫy lỗi công thức cho các công thức logic trong báo biểu. Bạn cũng sẽ học cách sử dụng lại mã công thức để tạo ra những hàm tùy biến.

Trong phần thứ hai của chương, bạn sẽ học cách sử dụng các câu lệnh SQL và biểu thức trong báo biểu của mình. Các lệnh SQL cho phép bạn sử dụng câu lệnh SQL như nguồn dữ liệu cho báo biểu và tận dụng nhiều hơn sức mạnh xử lý của máy chủ cơ sở dữ liệu.

2. GIỚI THIỆU VỀ CÔNG THỨC

Cho đến bây giờ, bạn đã phần nào quen thuộc với ngôn ngữ công thức của Crystal Reports, bạn đã thấy công thức hoạt động trong quá trình lọc mẫu tin, định dạng theo điều kiện. Tuy nhiên, ngoài việc sử dụng ngôn ngữ công thức, Crystal Reports còn có thể thực hiện những phép tính toán phức tạp khác trong báo biểu.

Nếu đã từng làm việc với một ngôn ngữ lập trình, ngôn ngữ công thức Crystal Reports sẽ trông rất quen thuộc. Có nhiều hàm thư viện của Crystal rất giống với các hàm trong Excel. Công thức trong Crystal Reports được viết và xây dựng thông qua trình Formula Editor (hình 5.1).



Hình 5.1: Hộp thoại công thức trong Crystal Reports.

Một công thức có thể bao gồm bất kỳ kiểu dữ liệu nào của bảng dữ liệu, kiểu số, toán tử, hàm, đoạn văn bản, các cấu trúc điều khiển, ví dụ như cấu trúc: **If ... Then**.

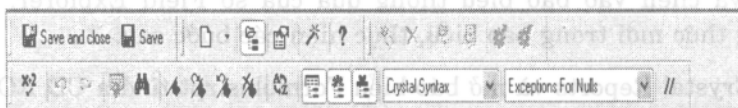
Trước tiên chúng ta sẽ tìm hiểu cách đưa công thức vào báo biểu và học cách sử dụng hộp thoại Formula Editor.

2.1. Sử dụng Formula Editor

Formula Editor là công cụ bạn có thể sử dụng để thêm hoặc sửa những công thức sẽ xuất hiện trong báo biểu. Để mở hộp thoại Formula Editor, bạn thực hiện các bước sau:

1. Mở báo biểu bạn đang làm việc.
2. Click View | Field Explorer.
3. Xác định mục Formula Editor, click phải vào tiêu đề mục chọn và chọn New.
4. Bạn sẽ được yêu cầu nhập tên cho công thức. Nhập vào tên và click OK. Hộp thoại Formula Editor trong Crystal Reports xuất hiện.
5. Hộp thoại Formula Editor bao gồm năm phần chính:

- **Workshop Tree:** Nằm bên cạnh trái của trang, chức năng này cho phép bạn định hướng và truy cập vào công thức từ bất kỳ nơi nào trong báo biểu.
- **Toolbar:** Chứa các biểu tượng để tạo công thức mới, nó cho phép bạn chuyển đổi giữa các công thức, tìm kiếm và thay thế nội dung công thức (hình 5.2).



Hình 5.2: Thanh công cụ trong hộp thoại Formula Editor.

- **Report Field Section:** Liệt kê tất cả các Field hiện diện trong báo biểu, nguồn dữ liệu và các bảng.
- **Operators list:** Danh sách các toán tử sẵn có trong Crystal Reports (như toán tử số học, toán tử so sánh v.v...).

Ghi chú: Thay vì gõ tay bạn có thể Double click vào bất kỳ toán tử nào trong danh sách để thêm nó vào trong báo biểu của mình.

- **Functions Section:** Liệt kê tất cả các hàm thư viện của Crystal.

2.2. Cú pháp Basic và cú pháp Crystal

Bạn có thể sử dụng đồng thời hai loại cú pháp để xây dựng công thức, đó là cú pháp của Crystal và cú pháp Basic (tương tự ngôn ngữ

Visual Basic). Cú pháp Crystal đã có từ rất lâu, trong các phiên bản trước đây cú pháp này là sự lựa chọn duy nhất cho những người phát triển báo biểu. Cú pháp Crystal Reports không có mối quan hệ trực tiếp tới bất kỳ ngôn ngữ lập trình nào (mặc dù nó tương tự như Pascal). Với phiên bản Crystal Reports 8.0, ngôn ngữ Basic mới được thêm vào. Trong Crystal Reports XI, cả hai kiểu cú pháp đều có thể sử dụng để xây dựng công thức trong cùng một báo biểu.

Để chọn cú pháp khi xây dựng công thức, bạn chỉ định trong danh sách xổ xuống ở góc phải của hộp thoại Formula Editor và chọn cú pháp thích hợp.

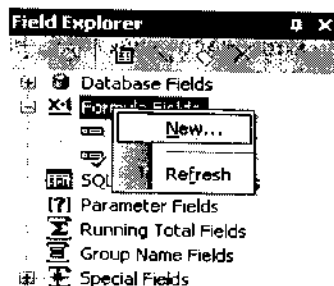
Nếu thật sự bắt đầu với Crystal Reports thì nên sử dụng cú pháp Crystal trước, bởi vì những công thức lọc mẫu tin trong Crystal Reports được viết chỉ để sử dụng cho cú pháp này. Tuy nhiên, Basic dễ sử dụng hơn nếu có kinh nghiệm về lập trình.

3. CÁCH HOẠT ĐỘNG CỦA CÔNG THỨC

3.1. Tạo một công thức mới

Giống như các Field trong Crystal Reports, các Field công thức có thể tạo và chèn vào báo biểu thông qua cửa sổ Field Explorer. Để tạo một công thức mới trong báo biểu, thực hiện các bước sau:

1. Mở Crystal Reports và mở báo biểu Formulas.rpt từ đĩa CD ROM.
2. Click View | Field Explorer, bạn sẽ thấy mục tên Formula Fields.
3. Để chèn một công thức mới, click phải vào mục Formula Fields và chọn New từ menu tắt, xem hình 5.3.



Hình 5.3: Tạo công thức mới từ hộp thoại Field Explorer.

4. Tiếp theo hộp thoại xuất hiện, nhập vào tên cho công thức và click OK. Có thể đặt bất cứ tên gì có ý nghĩa với bạn, trong ví dụ này, tên của công thức ta đặt là *OrderWithDiscount*.

Ghi chú: Nếu muốn tạo ra nhiều công thức, bạn có thể đặt một vài tên theo quy ước (ví dụ: *SalesTax1*, *SalesTax2*, v.v...) cho tên công thức của mình.

5. Hộp thoại Formula Editor xuất hiện, bạn có thể nhập vào đoạn công thức. Mở rộng mục Report Fields, và mở bảng Orders, double click vào Field **Orders.Order Amount** để chèn công thức.
6. Tiếp theo, cách một khoảng trắng, nhập dấu trừ (-) và nhập 200, bây giờ công thức sẽ như sau:

`{Orders.Order Amount} -200`

Ghi chú: Lúc này, bạn có thể kiểm tra công thức bằng cách click vào nút *X+2* trên thanh công cụ, nhưng hãy nhớ rằng *Crystal Reports* luôn kiểm tra cú pháp bất cứ khi nào bạn thoát ra khỏi Formula Editor.

7. Khi đã hoàn thành việc tạo mới công thức, click Save và đóng hộp thoại Formula Editor. Quay trở về môi trường thiết kế, bạn có thể kéo thả, chèn công thức vào trong báo biểu từ cửa sổ Field Explorer.

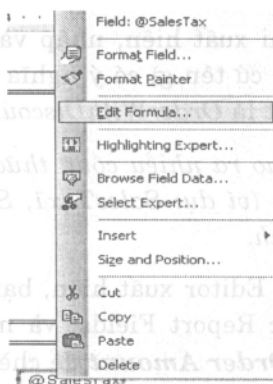
3.2. Chèn công thức vào trong báo biểu

Để chèn Field công thức bạn đã tạo ra, thực hiện các bước sau:

1. Trong hộp thoại Field Explorer (hiển thị bằng cách chọn View | Field Explorer), công thức của bạn sẽ được liệt kê dưới mục Formula Field.
2. Từ hộp thoại Field Explorer, kéo và thả Field công thức vào trong phần chi tiết trên báo biểu.
3. Bạn có thể click chọn Field và nhấn phím Enter, sau đó click trở chuột vào bất kỳ nơi nào trên báo biểu mà bạn muốn đặt công thức vào đó.

3.3. Sửa công thức đã tồn tại

Crystal Reports nhận dạng Field công thức mà bạn chèn vào trong báo biểu bởi ký hiệu @ đặt phía trước tên công thức và được bao quanh bởi dấu ngoặc {}. Ví dụ như tên Field là *OrderWithDiscount*, công thức sẽ xuất hiện như sau `{@OrderWithDiscount}`.



Hình 5.4: Để sửa công thức, click chuột phải vào tên công thức.

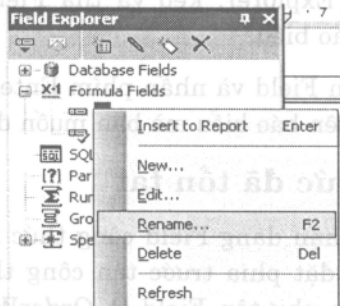
Bạn có thể sửa bất kỳ công thức nào mà bạn đã chèn vào trong báo biểu, bằng cách xác định Field công thức muốn sửa và click phải vào công thức đó. Sau đó chọn Edit Formula từ menu tắt, xem hình 5.4. Hộp thoại Formula Editor trong Crystal Reports sẽ xuất hiện giúp bạn có thể sửa nội dung Field công thức.

Khi đã hoàn tất việc sửa nội dung công thức, click nút Save và click nút Close để đóng hộp thoại Formula Editor.

3.4. Đổi tên công thức

Thỉnh thoảng bạn muốn xem lại báo biểu thiết kế và sửa đổi tên công thức, Field tham số để người thiết kế hoặc người sử dụng dễ hiểu hơn.

Để đổi tên một công thức, click View | Field Explorer. Cửa sổ Field Explorer sẽ hiển thị, bung rộng mục Formula Field và tìm công thức muốn đổi tên. Nếu click phải vào tên công thức, bạn sẽ thấy tùy chọn Rename trên menu tắt, (hình 5.5). Chọn Rename để sửa tên công thức.



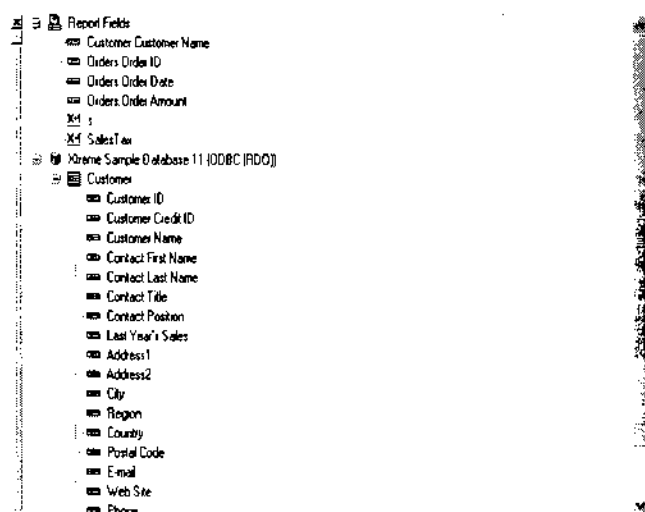
Hình 5.5: Bạn có thể đổi tên công thức từ hộp thoại Field Explorer.

Ghi chú: Khi đổi tên công thức sẽ được cập nhật ở tất cả những nơi nó đang sử dụng trong báo biểu.

3.5. TẠO CÁC CÔNG THỨC SỐ HỌC ĐƠN GIẢN

Những toán tử số học đơn giản như (+, -, *, /) được sử dụng trong công thức và dùng tính toán. Để tạo một công thức số học đơn giản, bạn thực hiện theo các bước sau:

1. Mở Crystal Reports và mở báo biểu Formulas.PRT mà chúng ta đang làm việc hoặc copy thêm một báo biểu Formulas.PRT từ CD ROM vào ổ cứng.
2. Click View | Field Explorer mở cửa sổ Field Explorer, click phải vào tên mục Formula Field và chọn New từ menu tắt.
3. Trong hộp thoại tiếp theo, bạn nhập vào tên Field công thức - đặt tên là SalesTax và click OK. Hộp thoại Formula Editor sẽ xuất hiện, cho phép bạn nhập vào nội dung công thức.
4. Trong ví dụ này, chúng ta cần dùng Field Order Amount (hình 5.6), Field Last Year Sales. Các Field này nằm trong mục Report Fields.



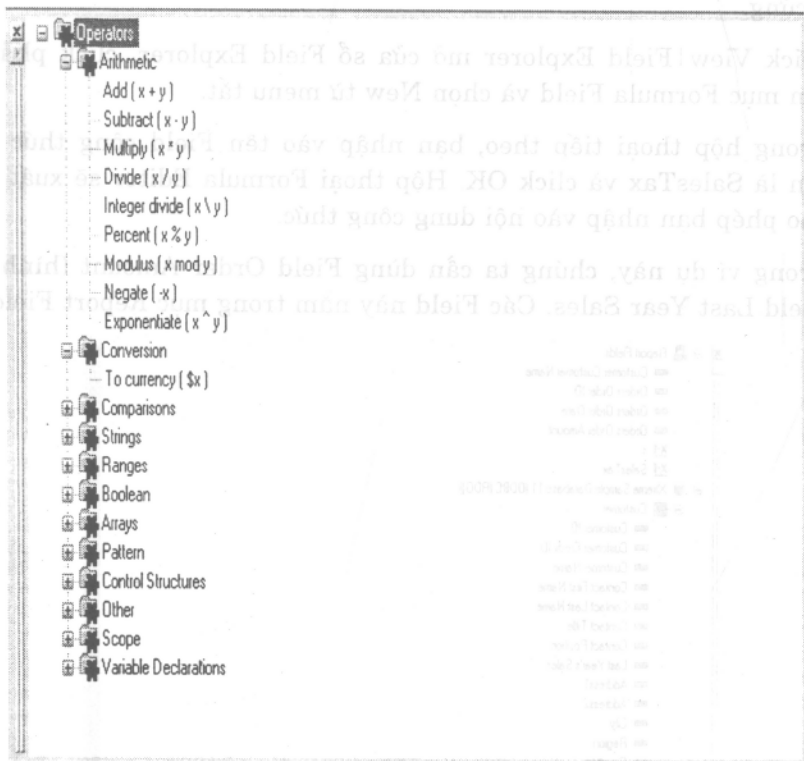
Hình 5.6: Danh sách Report Fields.

5. Để chọn một Field cần sử dụng trong nội dung công thức, double click vào tên Field. Field này xuất hiện trong nội dung công thức ở bên dưới. Trong ví dụ này, double click vào Field Orders.Order Amount.

Trong hộp thoại Formula Editor bạn có thể chọn hàm, các toán tử bằng cách double click bất kỳ mục nào của danh sách và nó sẽ được chèn vào trong nội dung công thức.

- Trong công thức này, chúng ta sẽ nhân Field Order Amount, với tỉ lệ 6% và sau đó cộng thêm \$20. Để làm điều này, hãy di chuyển tới khung Operators.

Khung Operators, xem hình 5.7, chứa tất cả các toán tử trong Crystal Syntax. Những toán tử này được phân loại theo toán tử số học, toán tử chuyển đổi, v.v...



Hình 5.7: Operators hiển thị danh sách tất cả các toán tử sẵn có trong Crystal Reports.

- Trong trường hợp này, do phép nhân là toán tử số học, bạn có thể chọn mục Arithmetic, xác định toán tử phép nhân (*) và sau đó double click vào nó để chèn vào trong nội dung công thức.

8. Bây giờ việc chúng ta cần làm là đưa vào giá trị 1.06, để cho thấy số lượng thêm vào 6%. Công thức như sau:

$\{Orders.Order Amount\} * 1.06$

9. Tiếp theo, chúng ta cần thêm \$20 vào công thức.

$(\{Orders.Order Amount\} * 1.06) + 20$

Vậy là xong. Nếu đặt công thức này vào trong phần Details của báo biểu, kết quả sẽ như hình 5.8.

Formulas				
Customer Name	Order ID	Order Date	Order Amount	Sales Tax
Deals on Wheels	1002	12/02/2003	\$5,060.28	\$5,383.90
Warsaw Sports, Inc.	1003	12/02/2003	\$186.87	\$198.08
Bikes and Trikes	1004	12/02/2003	\$823.05	\$892.43
SAB Mountain	1005	12/03/2003	\$29.00	\$50.74
Poser Cycles	1006	12/03/2003	\$64.90	\$88.79
Spokes	1007	12/03/2003	\$49.50	\$72.47
Clean Air Transportation Co.	1008	12/03/2003	\$2,214.94	\$2,367.84
Extreme Cycling	1009	12/03/2003	\$29.00	\$50.74
Cyclopath	1010	12/03/2003	\$14,872.30	\$15,784.64
BBS Ply	1011	12/03/2003	\$29.00	\$50.74
Piccolo	1012	12/03/2003	\$10,259.10	\$10,894.65
Pedals Inc.	1013	12/03/2003	\$1,142.13	\$1,230.66
Spokes 'N' Wheels Ltd	1014	12/04/2003	\$29.00	\$50.74
Cycle City Rome	1015	12/04/2003	\$43.50	\$66.11
SAB Mountain	1016	12/04/2003	\$563.70	\$617.52
Tyred Out	1017	12/05/2003	\$72.00	\$96.32
Has Been Bikes (consignme	1018	12/05/2003	\$115.50	\$142.43
Spokes for Folks	1019	12/05/2003	\$43.50	\$66.11
Extreme Cycling	1020	12/05/2003	\$67.80	\$91.87
Canal City Cycle	1021	12/06/2003	\$5,237.55	\$5,571.80
Reinnum Bike Co	1022	12/07/2003	\$2,797.86	\$2,980.47

Hình 5.8: Ví dụ công thức có tác động trên báo biểu.

3.6. Thứ tự toán tử trong Crystal Reports

Crystal Reports sử dụng thứ tự tính toán từ trái sang phải và ưu tiên:

- ☐ Dấu ngoặc.
- ☐ Số mũ.
- ☐ Phép nhân.
- ☐ Phép chia.
- ☐ Phép cộng.

Ví dụ, nếu tính toán số tuổi của một người nào đó theo ngày sinh dựa vào một Field trong cơ sở dữ liệu, bạn có thể sử dụng công thức sau:

`Today - {Staff.BirthDate}`

Vấn đề đối với công thức này là khi chèn vào báo biểu, nó sẽ là số ngày thay vì là số năm. Một giải pháp là bạn chia cho 365 ngày theo công thức sau:

`(Today - {Staff.BirthDate})/365`

Bạn nên nhớ thứ tự tính toán của các toán tử để khi thực hiện tính toán nên đặt toán tử nào xuất hiện trước trong khi làm việc.

4. SỬ DỤNG CÁC HÀM TRONG CRYSTAL REPORTS

Bạn có thể sử dụng các hàm thư viện của Crystal Reports để đơn giản hóa những tính toán phức tạp. Nếu mở rộng mục Functions của hộp thoại Crystal Reports Formula Editor, bạn sẽ thấy tất cả các hàm sẵn có, được sắp xếp theo chức năng. Để chèn một hàm vào nội dung công thức, bạn chỉ cần double click vào tên hàm.

Các hàm nối chung yêu cầu một hoặc nhiều đối số cách nhau bằng dấu phẩy. Khi chèn một hàm vào trong nội dung công thức, Crystal Reports tự động thêm dấu ngoặc đơn dấu phẩy để chỉ những đối số cần có, ví dụ khi bạn chọn hàm Round:

`Round ( , )`

Trong ví dụ này, bạn cần chỉ rõ một đối số cần làm tròn và số thập phân sẽ được sử dụng như sau:

`Round({Orders.Order Amount}, 2)`

Crystal Reports cung cấp hơn 200 hàm, bạn có thể chọn mục Help Crystal Reports (Crystal Reports online Help) để tham khảo chúng. Nhấn phím F1 (hoặc click Help|Crystal Reports Help) để hiển thị màn hình Crystal Reports Help như hình 5.9. Chọn tab Index và nhập vào tên hàm cần tìm.



Hình 5.9: Hình Help trong Crystal Reports.

Ghi chú: Ngoài các hàm sẵn có trong Crystal Reports, bạn cũng có thể tạo ra những hàm tùy biến cho chính mình, chúng ta sẽ học trong phần sau của chương này. Hàm này có thể được sử dụng lại trong báo biểu.

4.1. Các hàm tổng gộp

Bạn có thể sử dụng những hàm tổng gộp để tính tổng giá trị các Field xuất hiện trong báo biểu. Tổng gộp có thể là hàm tổng, có thể là hàm đếm hoặc tính trung bình cộng. Dưới đây là một số hàm tổng gộp phổ biến thường sử dụng.

Tên hàm	Cú pháp	Mô tả
Sum	Sum ()	Tính tổng cho một Field đặc biệt, ví dụ, Sum(Orders.Order Amount)).

Average	Avg()	Tính trung bình cho một Field đặc biệt, ví dụ <i>Avg(Orders.Order Amount)</i> .
Min	Min()	Tìm giá trị nhỏ nhất, ví dụ <i>Min(Orders.Order Amount)</i> trả về giá trị nhỏ nhất của Field <i>Order Amount</i> .
Max	Max()	Tìm giá trị lớn nhất, ví dụ <i>Max(Orders.Order Amount)</i> trả về giá trị lớn nhất của Field <i>Order Amount</i> .
Count	Count()	Đếm tất cả giá trị của Records, ví dụ <i>Count(Orders.OrderID)</i> .
Distinct Count	DistinctCount()	Đếm tất cả giá trị của Records trừ những giá trị trùng, nghĩa là mỗi giá trị giống nhau chỉ đếm một lần, ví dụ <i>DistinctCount(Orders.OrderID)</i> .

4.2. Hàm xử lý chuỗi

Crystal Reports cung cấp hai phép toán ghép nối chuỗi là: (+) và (&).

Toán tử (+) giống như toán tử cộng hai số, nhưng nó áp dụng tới những Field chuỗi, ví dụ:

```
{Customer.First Name} + " " + {Customer.Last Name}
```

Ghi chú: Nếu chuỗi là hằng số thì nó cần được bỏ trong cặp dấu nháy "", ví dụ ("Mr" + {customer.Last Name}).

Ký hiệu (&) là toán tử ghép nối chuỗi có thể được sử dụng như dấu cộng. Ví dụ:

```
{Customer.First Name} & " " & {Customer.Last Name}
```

Sự khác nhau giữa hai toán tử là ký hiệu, nhưng toán tử (&) thì linh hoạt hơn và có thể sử dụng để ghép nối những Field kiểu số và những Field khác (mà không phải chuyển đổi chúng sang field kiểu chuỗi). Ví dụ:

```
{Customer.First Name} & " " & {Customer.Last Name}
& {Customer.CustomerID}
```

Nếu bạn thực hiện điều này bằng toán tử dấu cộng (+), Crystal Reports sẽ phát sinh lỗi do Field CustomerID là Field kiểu số và toán tử (+) chỉ thực hiện với Field kiểu chuỗi.

Bảng dưới đây liệt kê một số hàm xử lý chuỗi thường sử dụng nhất.

Tên hàm	Kết quả	Cách sử dụng
Uppercase ()	<i>Chuyển tất cả các chuỗi thành chữ hoa.</i>	Uppercase ({Table.FiedName})
Lowercase ()	<i>Chuyển tất cả các chuỗi thành chữ thường.</i>	Lowercase ({Table.FiedName})
Length ()	<i>Đếm chiều dài của chuỗi.</i>	Length ({Table.FieldName})
Trim ()	<i>Xóa những ký tự trắng ở đầu và cuối của một chuỗi.</i>	Trim ({Table.FieldName})
Left (String, Length)	<i>Trả về số ký tự từ bên trái của chuỗi.</i>	Left ({Table.FieldName} , 5)
Right (String, Length)	<i>Trả về số ký tự từ bên phải của chuỗi.</i>	Right ({Table.FieldName} , 5)

4.3. Các hàm chuyển đổi kiểu

Khi sử dụng Crystal Reports, có thể bạn gặp những vấn đề liên quan đến kiểu dữ liệu của Field chứa trong cơ sở dữ liệu. Chẳng hạn chuyển đổi kiểu số sang chuỗi, chuyển ngày sang dạng số... Dưới đây là các hàm chuyển đổi cho các kiểu dữ liệu thông dụng.

Kiểu đích	Hàm	Kiểu dữ liệu nguồn
Text	ToText ()	Number, Currency, DateTime, Date, Time.
Number	ToNumber ()	String, DateTime, Date, Time.
Boolean	Cbool ()	Number, Currency.
Currency	Ccur ()	Number, Currency or String.
Date Time	CdateTime ()	Number, String, DateTime, Date, Time.
Integer	CDbl ()	Number, Currency, String or Boolean.
String	Cstr ()	Number, Currency, String, DateTime, Date, Boolean.
Date	Cdate ()	Number, Currency, String, DateTime, Time.

Ví dụ để chuyển giá trị kiểu số của Field Order Amount sang kiểu chuỗi, bạn gọi hàm ToText như sau:

```
ToText ( {Orders.Order Amount} )
```

4.4. Các hàm theo khoảng thời gian và Field ngày tháng

Crystal Reports có một số hằng định nghĩa trước xác định khoảng thời gian được sử dụng với ngày tháng. Bạn đã sử dụng qua các biến xác định khoảng thời gian ngày tháng để lọc mẫu tin.

Khi làm việc với những khoảng thời gian, Crystal Reports sẽ xử lý mọi công việc phức tạp về ngày tháng hộ bạn. Ví dụ, khi sử dụng khoảng thời gian *MonhTodate*, Crystal Reports sẽ kiểm tra ngày và sau đó xây dựng một danh sách tất cả các ngày nằm trong khoảng thời gian xác định của *MonhTodate* (đến một ngày chỉ định trong tháng).

Các giá trị về khoảng thời gian trong Formula Editor được sử dụng thường xuyên chung với toán tử *IN*. Ví dụ:

```
If {Order.OrderDate} in Over90days
    then " Past Due Account"
```

Bạn cũng có thể sử dụng kỹ thuật này với những khoảng thời gian ngày tháng khác như:

- WeekToDateFromSunday
- MothToDate
- YearToDate

- Last7Days
- Last4WeeksToSun
- LastFullWeek
- LastFullMonth
- AllDatesToYesterday
- AllDatesFromToDay
- AllDatesFromTomorrow
- Aged0To30Days
- Aged31To60Days
- Aged61To90Days
- Over90Days
- Next30Days
- Next31To60Days
- Next61To90Days
- Next91To365Days
- Calendar1stQtr
- Calendar2ndQtr
- Calendar3rdQtr
- Calendar4thQtr
- Calendar1stHalf
- Calendar2ndHalf

Ngoài ra để sử dụng những hàm thời gian với ngày tháng, Crystal Reports cho phép bạn sử dụng một số phép tính trên Field ngày tháng. Ví dụ, Crystal Reports cho phép bạn tính sự khác nhau giữa hai ngày cũng như thêm một số ngày vào một ngày nào đó, kết quả trả về là một giá trị ngày khác, ví dụ:

```
{Orders.ShipDate} - {Orders.OrderDate}
```

Muốn xem sau 30 ngày nữa là ngày mấy so với ngày giao hàng, bạn có thể viết:

```
{Orders.ShipDate} + 30
```

Ghi chú: Bạn có thể cộng và trừ ngày tháng, nhưng bạn không thể nhân hoặc chia chúng.

5. TẠO CÔNG THỨC LOGIC

Trong các ví dụ trước đây, bạn đã thấy phát biểu `If...then`. Phát biểu này lựa chọn điều kiện đúng hay sai để quyết định chọn kết quả.

Để tìm hiểu phát biểu `If...then` hoạt động ra sao, chúng ta sẽ xem qua ví dụ sau: nếu muốn hiển thị tất cả mẫu tin của Field Order Amount có giá trị trên 1000, chúng ta có thể sử dụng công thức `If...then`:

If {Orders.Order Amount}>1000 then " Great Sale"

Nếu chúng ta đặt công thức này gần Field Order Amount trên phần chi tiết của báo biểu, thì công thức này sẽ được ước lượng khi xử lý mẫu tin trong bảng. Nếu điều kiện đúng công thức sẽ xuất hiện thông báo "Great Sale" và ngược lại, nếu điều kiện sai sẽ không xuất hiện thông báo, xem hình 5.10.

Formulas				
Customer Name	Order ID	Order Date	Order Amount	Sales Tax
Deals on Wheels	1002	12/02/2003	\$5,080.28	Great Sale!
Warsaw Sports, Inc	1003	12/02/2003	\$186.97	
Bikes and Trikes	1004	12/02/2003	\$823.05	
SAB Mountain	1005	12/03/2003	\$29.00	
Poser Cycles	1006	12/03/2003	\$64.90	
Spokes	1007	12/03/2003	\$49.50	
Clean Air Transportation Co	1008	12/03/2003	\$2,214.94	Great Sale!
Extreme Cycling	1009	12/03/2003	\$29.00	
Cyclopath	1010	12/03/2003	\$14,872.30	Great Sale!
BBS Ply	1011	12/03/2003	\$29.00	
Piccolo	1012	12/03/2003	\$10,259.10	Great Sale!
Pedals Inc	1013	12/03/2003	\$1,142.13	Great Sale!
Spokes 'N' Wheels Ltd.	1014	12/04/2003	\$29.00	
Cycle City Rome	1015	12/04/2003	\$43.50	
SAB Mountain	1016	12/04/2003	\$563.70	
Tyred Out	1017	12/05/2003	\$72.00	
Has Been Bikes (consignme	1018	12/05/2003	\$115.50	
Spokes for Folks	1019	12/05/2003	\$43.50	
Extreme Cycling	1020	12/05/2003	\$67.60	
Caral City Cycle	1021	12/06/2003	\$5,237.55	Great Sale!
Deals on Wheels	1022	12/07/2003	\$3,202.00	Great Sale!

Hình 5.10: Một ví dụ về sử dụng công thức If...then

Với phát biểu **If...then**, chúng ta cũng có thể thêm một phát biểu **Else** vào sau, phát biểu **Else** sẽ có hiệu lực khi điều kiện **If** không đúng.

Ví dụ, ta có thể viết lại công thức trên như sau:

**If {Orders.Order Amount}>1000 then " Great Sale"
else "Good Sale"**

Trong trường hợp này, nếu điều kiện là đúng (giá trị Field Order Amount lớn hơn 1000), mệnh đề đầu sẽ được thực hiện, in ra "Great Sale" trên báo biểu; mặt khác, nếu điều kiện là sai (giá trị Field Order Amount nhỏ hơn 1000), thông báo "Good Sale" sẽ được in ra trong báo biểu.

Phát biểu **If...then** có thể được kết hợp với những hàm và toán tử logic khác như **AND**, **OR** để tạo ra những công thức phức tạp.

6. THỜI ĐIỂM ƯỚC LƯỢNG CÔNG THỨC

Crystal cung cấp các phân đoạn và các vị trí khác nhau của báo biểu. Đặt công thức trong những phân đoạn báo biểu khác nhau sẽ cho kết quả khác nhau. Bạn cần phải hiểu rõ khi nào thì công thức được đem ra ước lượng (tính toán). Crystal có thể đem công thức ra tính toán ở 4 thời điểm

- ❖ **BeforeReadingRecords**
- ❖ **WhileReadingRecords**
- ❖ **WhilePrintingRecords**
- ❖ **EvaluateAfter** (Tên công thức)

Mỗi hàm xác định thời gian tính toán này có thể được sử dụng kết hợp với nội dung công thức và xác định lúc nào thì công thức sẽ được xử lý khi báo biểu thực thi. Ví dụ:

```
WhilePrintingRecords;
    NumberVar SalesCount;
    SalesCount := SalesCount + 1;
    SalesCount;
```

Phần dưới đây ta sẽ tìm hiểu ý nghĩa của từng hàm ước lượng thời gian tính toán.

6.1. BeforeReadingRecords

Hàm *BeforeReadingRecords* sẽ được ước lượng trước khi đọc dữ liệu mẫu tin. Quá trình ước lượng này được áp dụng cho các hằng, biến không sử dụng đến dữ liệu mẫu tin và thường chỉ diễn ra 1 lần trong suốt quá trình báo biểu thực thi.

6.2. WhileReadingRecords

Hàm *WhileReadingRecords* sẽ ước lượng khi Crystal duyệt qua dữ liệu từng mẫu tin trong bảng dữ liệu. Thường nếu có các phép tính cộng dồn hay xem xét chuyển đổi giá trị của các Field trong cơ sở dữ liệu (tức là, thay đổi từ kiểu chuỗi sang kiểu số), bạn có thể sử dụng hàm ước lượng này như sau:

```
WhileReadingRecords;
    ToText ({Orders.Order Amount})
```

6.3. WhilePrintingRecords

Hàm **WhilePrintingRecords** là hàm ước lượng phổ biến nhất. Hàm này được ước lượng khi mẫu tin kết xuất ra trang in hay Preview trên báo biểu. Hàm này có thể được sử dụng như tính tổng chạy, hoặc đếm số lượng v.v... Ví dụ:

```
WhilePrintingRecords:
    NumberVar StatusCounter:
    If {Customer.status}="Overdue" then
        StatusCounter:=StatusCounter+1;
    StatusCounter;
```

CD-ROM: Xem ví dụ báo biểu Mantotal.rpt trong thư mục Project trên CD ROM để hiểu rõ hơn về hàm **WhilePrintingRecords**.

6.4. EvaluateAfter

Hàm **EvaluateAfter** được gọi sau khi kết thúc xử lý một công thức. Ví dụ, nếu có công thức tính là Order Total, bạn muốn đặt công thức Sales Tax sau công thức Order Total, như vậy bạn có thể đảm bảo công thức Order Total được tính trước công thức Sales Tax.

Trong CD ROM bao gồm những ví dụ mẫu sử dụng tất cả ba hàm ước lượng trên. Bạn tìm file Evaltime.Zip trong thư mục Projects.

7. BẦY LỖI CÔNG THỨC

Khi sử dụng công thức, bạn phải chắc chắn rằng cú pháp mà bạn đưa vào công thức là đúng. Trong Crystal Reports có một chức năng giúp bạn kiểm tra cú pháp của công thức.

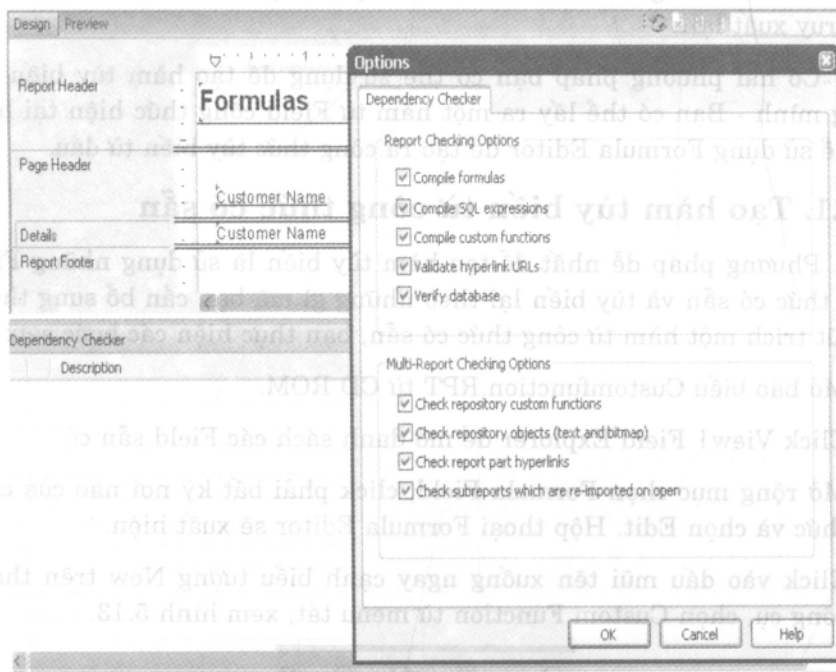
Khi lưu công thức bằng cách click vào biểu tượng Save trên thanh công cụ, xem hình 5.11, Crystal Reports tự động thực hiện sự kiểm tra cú pháp, để chắc chắn rằng công thức không thiếu dấu ngoặc đơn, các phép toán cộng trừ v.v... Bạn cũng có thể click vào biểu tượng X+2 khi đang làm việc trên trong hộp thoại Formula Editor.



Hình 5.11: Kiểm tra công thức của bạn trước khi thoát khỏi Formula Editor.

Nếu việc kiểm tra cú pháp trả về lỗi, con trỏ chuột sẽ được di chuyển tới vị trí trong công thức nơi lỗi xuất hiện để bạn sửa đổi lại nội dung công thức.

Crystal Reports XI cung cấp thêm công cụ Dependency Checker, xem hình 5.12.



Hình 5.12: Hoạt động Dependency Checker.

Khi cần giúp đỡ, Dependency Checker sẽ chạy thử báo biểu của bạn và tìm ra các lỗi cũng như xác định những phần phụ thuộc của công thức còn thiếu. Ví dụ, nếu báo biểu có một công thức liên kết với nhóm tính tổng gộp, nếu bạn xóa nhóm, công thức sẽ xuất hiện trong Dependency Checker như một lỗi, bạn cần đóng công thức đó lại trước khi chạy báo biểu. Tương tự, nếu có một Field bị thiếu hoặc tên không đúng, Dependency Checker sẽ kiểm tra điều này và phát hiện lỗi nhanh chóng.

Ghi chú: Bạn có thể hiển thị tùy chọn cho Dependency Checker bằng cách click phải trên thanh công cụ. Chọn Options từ menu tắt.

Với cả hai phương pháp trên, công thức của bạn sẽ luôn tính toán và trả về giá trị đúng.

8. LÀM VIỆC VỚI HÀM TÙY BIẾN

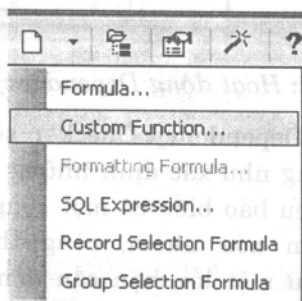
Crystal Reports cho phép bạn viết các hàm tùy biến. Hàm tùy biến đơn giản là một Field công thức mà bạn có thể lưu trữ và sử dụng lại trong các báo biểu. Ngoài ra, nếu đang sử dụng BusinessObjects Enterprise, bạn có thể lưu công thức vào kho chứa Repository để các báo biểu khác có thể truy xuất lại.

Có hai phương pháp bạn có thể sử dụng để tạo hàm tùy biến của riêng mình - Bạn có thể lấy ra một hàm từ Field công thức hiện tại hoặc có thể sử dụng Formula Editor để tạo ra công thức tùy biến từ đầu.

8.1. Tạo hàm tùy biến từ công thức có sẵn

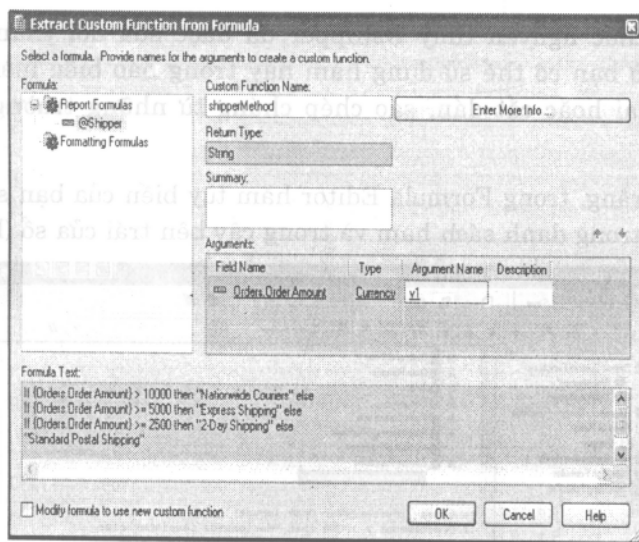
Phương pháp dễ nhất để tạo hàm tùy biến là sử dụng những Field công thức có sẵn và tùy biến lại theo những gì mà bạn cần bổ sung thêm. Để rút trích một hàm từ công thức có sẵn, bạn thực hiện các bước sau:

1. Mở báo biểu Customfunction.RPT từ CD ROM.
2. Click View | Field Explorer để mở danh sách các Field sẵn có.
3. Mở rộng mục chọn Formula Field, click phải bất kỳ nơi nào của công thức và chọn Edit. Hộp thoại Formula Editor sẽ xuất hiện.
4. Click vào dấu mũi tên xuống ngay cạnh biểu tượng New trên thanh công cụ, chọn Custom Function từ menu thả, xem hình 5.13.



Hình 5.13: Tạo mới công thức tùy biến.

5. Nhập vào tên của công thức tùy biến, trong ví dụ này, chúng ta sẽ gọi tên công thức là *ShipMethod*.
6. Sau khi nhập tên hàm tùy biến xong, click vào nút **Use Extract**, hộp thoại Extract Custom Function sẽ xuất hiện, xem hình 5.14.



Hình 5.14: Hộp thoại *Extract Custom Function*

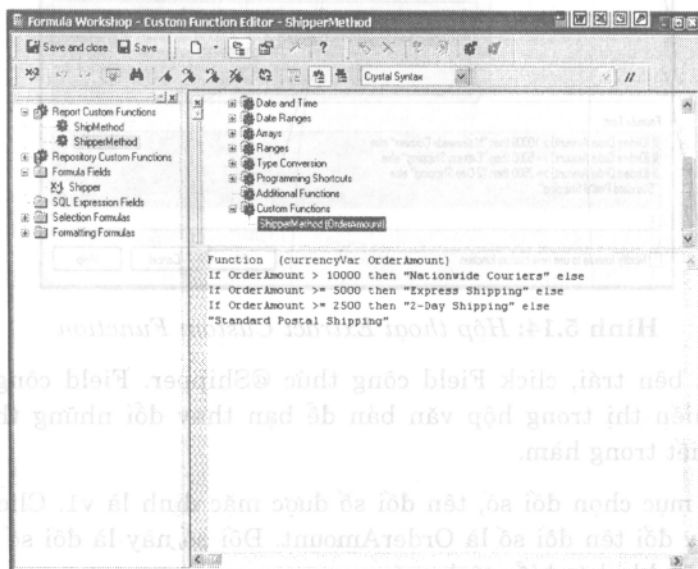
7. Từ cột bên trái, click Field công thức @Shipper. Field công thức sẽ được hiển thị trong hộp văn bản để bạn thay đổi những thuộc tính cần thiết trong hàm.
8. Trong mục chọn đối số, tên đối số được mặc định là v1. Click vào v1 để thay đổi tên đối số là OrderAmount. Đối số này là đối số được yêu cầu nhập khi báo biểu tính toán.
9. Bạn chú ý có một nút trên góc phải có tên **Enter More Info**. Nếu chọn nút **Enter More Info**, trang thuộc tính mới sẽ hiển thị, khi đó bạn có thể thêm nội dung mô tả hàm, tên tác giả và một vài đoạn hướng dẫn cho người sử dụng.
10. Tiếp theo, click vào checkbox ở góc dưới mang tên **Modify Formula to use new custom function** và click OK.
11. Xem lại nội dung công thức trong Text Formula, trong ví dụ của ta nội dung công thức như sau:

```
Function ( currencyVar OrderAmount )
    If OrderAmount>1000 then " Nationwide Coriers " else
    If OrderAmount >=5000 then "Express Shipping" else
    If OrderAmount >=2500 then "2 day shipping"
    else "Standard Postal Shipping"
```

12. Click vào nút Save và Close để quay lại cửa sổ thiết kế báo biểu.

Công thức nguyên thủy @shipper đã được sửa đổi thành hàm tùy biến. Bây giờ bạn có thể sử dụng hàm này trong báo biểu mà không cần định nghĩa lại hoặc cắt dán, sao chép chúng từ những trường công thức khác.

Lưu ý rằng, trong Formula Editor hàm tùy biến của bạn sẽ xuất hiện trong cả hai trong danh sách hàm và trong cây bên trái cửa sổ (hình 5.15).



Hình 5.15: Những hàm quen thuộc sẽ xuất hiện trong Formula Editor.

Để sửa công thức, bạn double click vào cây danh sách bên trái của sổ để mở công thức cần sửa.

8.2. Tự tạo hàm tùy biến

Nếu không có bất kỳ công thức nào trên báo biểu để bạn sử dụng hay làm mẫu, bạn có thể tạo ra những hàm tùy biến từ đầu để tạo hàm tùy biến của riêng mình:

1. Mở báo biểu Functionformula.RPT từ CD ROM.
2. Click View | Field Explorer để mở danh sách các Field có sẵn.
3. Mở rộng mục chọn Formula Editor và click bất kỳ công thức nào. Chọn Edit từ thực menu tắt. Hộp thoại Formula Editor sẽ xuất hiện.
4. Sử dụng mũi tên xuống bên cạnh biểu tượng New, chọn Custom Function, hộp thoại xuất hiện nhắc bạn nhập vào tên hàm. Ở ví dụ

này bạn nhập tên hàm là *CountryGrouping*, sau đó click vào nút Use Editor. Hộp thoại Formula Editor sẽ xuất hiện.

5. Nhập đoạn công thức sau:

```
FUNCTION ( stringVar Country)
If Country ="USA" then "Domestic" else
If (Country ="Mexico" or Country ="Canada"
then "North America" else "Rest of world"
```

6. Khi hoàn thành, click vào nút Save và Close trên góc trái của màn hình.

7. Để tạo một công thức sử dụng hàm này, click View | Field Explorer.

8. Click phải vào mục Formula Field và chọn New từ menu tắt.

9. Nhập vào tên công thức, trong ví dụ này, nhập tên công thức là Country Flag.

10. Tiếp theo, nhập nội dung công thức sau:

```
CountryGrouping ( {Custom.Country} )
```

11. Khi hoàn thành, click vào nút Save và Close trên góc trái để dừng thiết kế báo biểu.

12. Từ Field Explorer, kéo Field công thức này vào trong phần chi tiết của báo biểu. Báo biểu sau khi thực thi sẽ như hình 5.16.

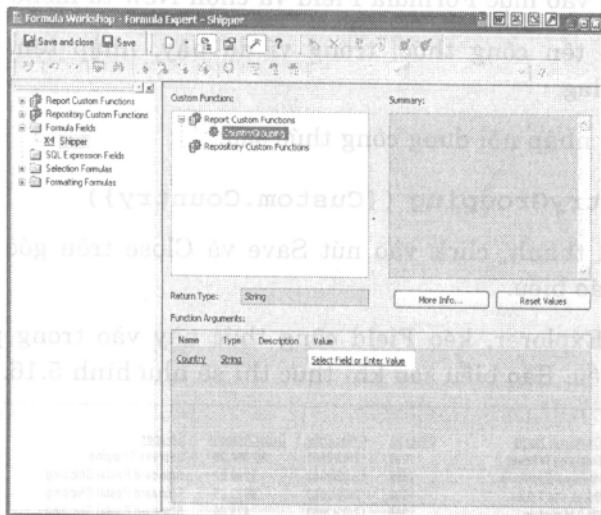
Customer Name	Order ID	Order Date	Order Amount	Shippe:
Deals on Wheels	1002	12/03/2003	\$5,060.28	Express Shipping
Warsaw Sports, Inc	1003	12/03/2003	\$186.87	Standard Postal Shipping
Bikes and Trikes	1004	12/03/2003	\$823.05	Standard Postal Shipping
SAB Mountain	1005	12/03/2003	\$29.00	Standard Postal Shipping
Poser Cycles	1006	12/03/2003	\$64.90	Standard Postal Shipping
Spokes	1007	12/03/2003	\$46.50	Standard Postal Shipping
Clean Air Transportation Co.	1008	12/03/2003	\$2,214.94	Standard Postal Shipping
Extreme Cycling	1009	12/03/2003	\$25.60	Standard Postal Shipping
Cyclopath	1010	12/03/2003	\$14,872.38	Nationwide Couriers
BBS Ply	1011	12/03/2003	\$29.60	Standard Postal Shipping
Pictolo	1012	12/03/2003	\$10,256.10	Nationwide Couriers
Pedals Inc.	1013	12/03/2003	\$1,142.13	Standard Postal Shipping
Spokes 'N' Wheels Ltd	1014	12/04/2003	\$29.00	Standard Postal Shipping
Cycle City Rome	1015	12/04/2003	\$42.50	Standard Postal Shipping
SAB Mountain	1016	12/04/2003	\$563.70	Standard Postal Shipping
Tyrod Jnr	1017	12/05/2003	\$12.00	Standard Postal Shipping
Has Been Bikes (consigne)	1018	12/05/2003	\$415.50	Standard Postal Shipping
Spokes for Folks	1019	12/05/2003	\$42.50	Standard Postal Shipping
Extreme Cycling	1020	12/05/2003	\$67.80	Standard Postal Shipping
Canal City Cycle	1021	12/06/2003	\$5,237.55	Express Shipping
Belgium Bike Co	1022	12/07/2003	\$2,792.86	2-Day Shipping
Tienda de Bicicletas El Pat	1023	12/07/2003	\$890.61	Standard Postal Shipping
Tandem Cycle	1024	12/07/2003	\$33.00	Standard Postal Shipping

Hình 5.16: Báo biểu đã hoàn thành.

8.3. Sử dụng các hàm tùy biến trong công thức

Hàm tùy biến hoàn toàn giống như những hàm thư viện khác mà bạn thấy trong Crystal Reports, đơn giản bạn có thể thêm chúng vào đoạn công thức của mình theo cú pháp thông dụng (Function(parameter1,parameter2,v.v...)). Ví dụ dưới đây sẽ hướng dẫn bạn cách sử dụng công thức:

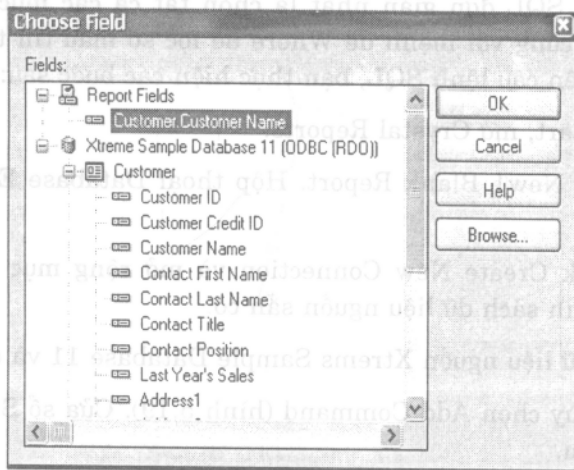
1. Mở báo biểu Functions wizard.RPT từ CD ROM.
2. Click View | Field Explorer để mở danh sách các trường sẵn có.
3. Mở rộng mục Formula Field, click phải và chọn New, nhập tên công thức. Hộp thoại Formula Editor sẽ xuất hiện.
4. Từ thanh công cụ, click trên biểu tượng Magic Wand (hình 5.17).



Hình 5.17: Function Expert.

5. Mở rộng danh sách Report Custom Functions và click vào hàm *CountryGrouping*. Góc dưới của hộp thoại hiển thị giá trị tham số.
6. Click vào liên kết **Select Field hoặc Enter Value** trên lưới. Bạn nhập vào giá trị hoặc sử dụng danh sách xổ xuống để chọn Field khác, hộp thoại chọn Field khác sẽ xuất hiện như hình 5.18.
7. Chọn một trường từ danh sách và click OK.
8. Tiếp theo, click nút Save và Close để quay trở về môi trường thiết kế báo biểu.

9. Bây giờ bạn có thể kéo Field công thức vào trong báo biểu. Chọn chế độ Preview để xem kết quả.



Hình 5.18: Chọn một Field.

9. CÂU LỆNH SQL VÀ BIỂU THỨC

Một trong những đặc tính mạnh nhất của Crystal Reports là khả năng cho phép bạn viết câu lệnh SQL cho báo biểu, như chọn bảng, kết nối, Field v.v... Trong những phiên bản trước của Crystal Reports, chức năng này dễ sử dụng cho người mới bắt đầu họ không biết cách viết lệnh SQL nhưng nó làm nản lòng người quản trị và người phát triển cơ sở dữ liệu vì không đủ mạnh. Trong phiên bản mới này, bạn có đầy đủ khả năng sửa đổi câu lệnh SQL và thêm viết những Field biểu thức SQL cần thiết.

9.1. Cách hoạt động của câu lệnh SQL

Câu lệnh SQL là gì? Trong Crystal Reports, đây là câu lệnh truy vấn và trả về bảng dữ liệu. Ví dụ, bạn có thể viết một câu lệnh SQL đơn giản sau để hiển thị kết quả từ bảng khách hàng trong cơ sở dữ liệu:

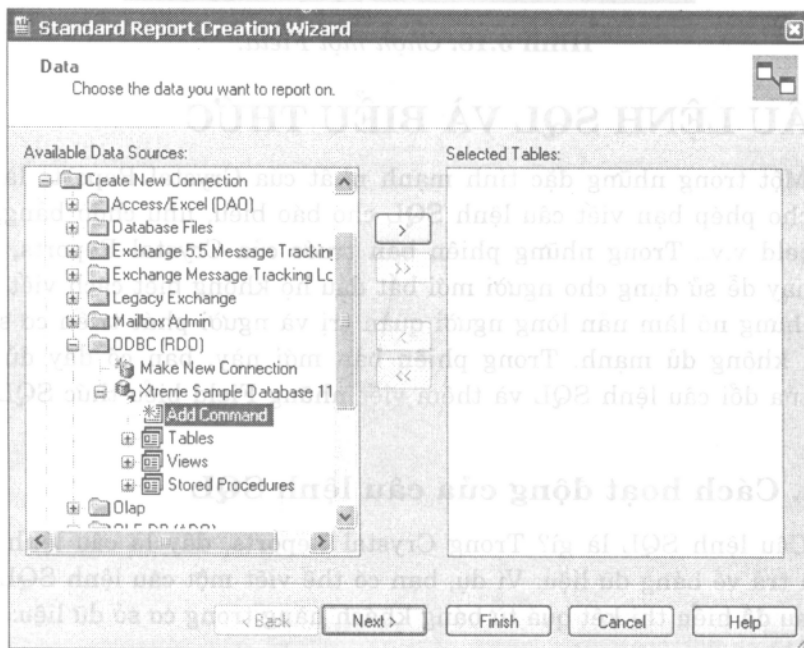
```
Select * from Customer
```

Khi câu lệnh SQL sử dụng trong báo biểu, tất cả các Field kết quả sẽ xuất hiện trong Field Explorer, và kết quả các mẫu tin trả về có thể được sử dụng như tập hợp dữ liệu cho báo biểu của bạn.

9.2. Tạo câu lệnh SQL

Câu lệnh SQL đơn giản nhất là chọn tất cả các mục từ một bảng (Table) dữ liệu cùng với mệnh đề Where để lọc số mẫu tin trả về. Để tạo báo biểu dựa trên câu lệnh SQL, bạn thực hiện các bước sau:

1. Từ menu Start, mở Crystal Reports.
2. Click File| New| Blank Report. Hộp thoại Database Expert sẽ xuất hiện.
3. Double click Create New Connection và mở rộng mục ODBC (RDO) để chọn danh sách dữ liệu nguồn sẵn có.
4. Chọn tên dữ liệu nguồn Xtrem Sample Database 11 và click Finish.
5. Click vào tùy chọn Add Command (hình 5.19). Cửa sổ SQL Command sẽ xuất hiện.

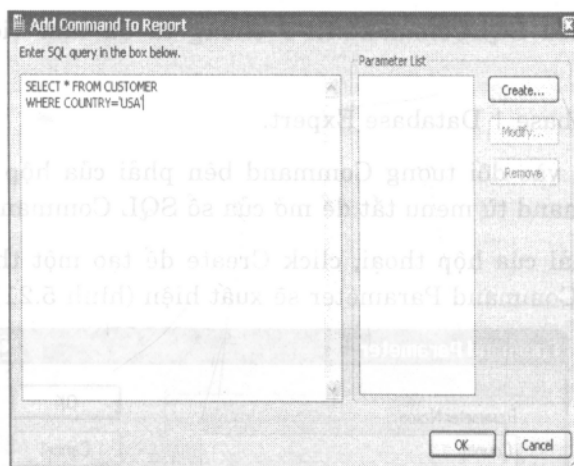


Hình 5.19: Tùy chọn Add Command xuất hiện dưới tên dữ liệu nguồn.

6. Nhập đoạn lệnh SQL vào trong hộp thoại:

Select * from Customer where Country="USA"

Bạn xem nội dung câu lệnh SQL trong hình 5.20.



Hình 5.20: Hộp thoại viết câu lệnh SQL.

7. Click OK để trở về hộp thoại Database Expert, sau đó click Finish để quay về cửa sổ thiết kế báo biểu.
8. Click View | Field Explorer để mở danh sách của các Field sẵn có. Các Field được liệt kê dưới mục Command. Bạn có thể kéo và thả chúng vào báo biểu.
9. Lưu báo biểu với tên SQLCommand.RPT.

Bạn có thể sử dụng lệnh SQL để lấy về một tập dữ liệu từ Table hoặc View trong cơ sở dữ liệu.

9.3. Sửa câu lệnh SQL

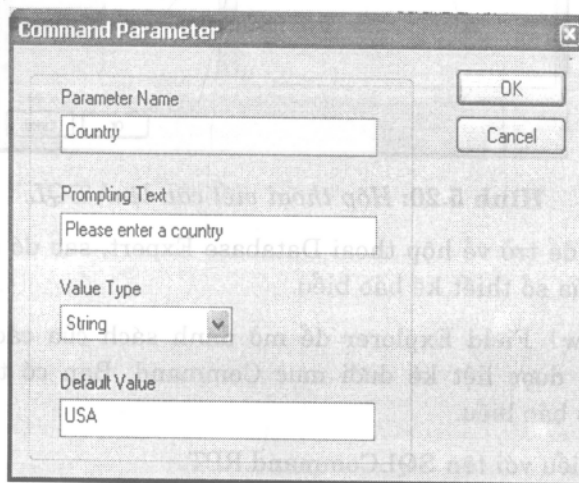
Trong Crystal Reports để sửa câu lệnh SQL bạn sử dụng hộp thoại Database Expert. Để mở hộp thoại Database Expert, click Database | Database Expert, sau đó click phải vào đối tượng Command. Trên menu tắt chọn Edit hoặc View cửa sổ SQL Command sẽ xuất hiện.

Từ cửa sổ SQL Command, bạn có thể sửa đổi nội dung câu lệnh SQL sau đó click OK để chấp nhận nội dung mới.

9.4. Thêm tham số vào lệnh SQL

Tham số là một tính năng mạnh trong Crystal Reports do chúng cho phép nhập và lọc thông tin khi chạy báo biểu. Với SQL Command, bạn có thể tạo những tham số cho câu lệnh SQL. Để thêm tham số từ câu lệnh SQL, bạn thực hiện các bước sau:

1. Mở báo biểu SQLCommand.RPT chúng ta đã làm việc trong phần trước .
2. Click Database | Database Expert.
3. Click phải vào đối tượng Command bên phải của hộp thoại và chọn Edit Command từ menu tắt để mở cửa sổ SQL Command.
4. Từ bên phải của hộp thoại, click Create để tạo một tham số mới và hộp thoại Command Parameter sẽ xuất hiện (hình 5.21).



Hình 5.21: Hộp thoại Command Parameter.

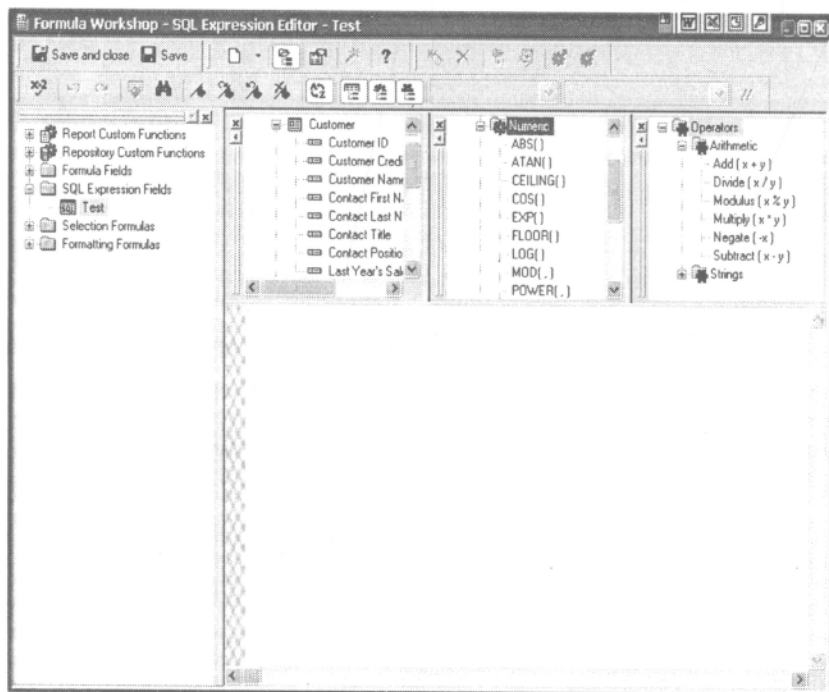
5. Nhập vào tên tham số, chuỗi thông điệp, kiểu giá trị và giá trị mặc định mà bạn muốn gán cho Field tham số. Trong ví dụ này, nhập tên tham số là Country, kiểu là String.
6. Khi hoàn thành, click OK để trở về cửa sổ SQL Command.
7. Tiếp theo, sửa đổi lại lệnh SQL Command cung cấp thêm tham số trong mệnh đề Where như sau:
Select * from Customer where Country='{?Country}'
8. Click OK để trở về hộp thoại Database Expert và click OK lần nữa để quay về môi trường thiết kế báo biểu.
9. Nhấn phím F5 để làm mới lại báo biểu. Khi ấy một thông báo nhắc bạn nhập vào giá trị tham số Country, tham số này sẽ được sử dụng để lọc báo biểu.

Một mấu chốt nên nhớ rằng các tham số trong SQL Command cũng giống như những tham số khác trong Crystal Reports (tức là, với một dấu hỏi được đặt trước tên Field sẽ được chứa trong dấu {}). Ví dụ như {?Country}.

9.5. Cách hoạt động của biểu thức SQL

Biểu thức SQL là phát biểu SQL được sử dụng giống như công thức và được lượng giá bởi cơ sở dữ liệu máy chủ thay vì Crystal Reports. Sự khác nhau giữa câu lệnh SQL và biểu thức SQL đó là biểu thức SQL được tính toán trên máy chủ. Để tạo một biểu thức SQL trong báo biểu, bạn thực hiện các bước sau:

1. Mở báo biểu SQLExpression.RPT trên CD ROM.
2. Mở Field Explorer bằng cách click View | Field Explorer, sau đó click phải vào mục SQL Expression Field và chọn New. Hộp thoại xuất hiện và nhắc bạn nhập tên biểu thức SQL.
3. Nhập tên và click OK, hộp thoại SQL Expression Editor sẽ xuất hiện (hình 5.22).



Hình 5.22: Hộp thoại Expression Editor.

4. Bạn sử dụng hộp văn bản phía dưới để soạn thảo, đưa vào biểu thức SQL.
5. Khi đã hoàn tất việc tạo biểu thức SQL, click nút Save để quay lại môi trường thiết kế báo biểu. Bây giờ bạn có thể kéo biểu thức SQL vào trong báo biểu như thường làm với các Field khác.

10. TÓM TẮT

Chương này đã hướng dẫn bạn cách xây dựng công thức cho báo biểu. Bạn có thể tạo các công thức tính toán và logic, Crystal cung cấp cho bạn ngôn ngữ lập trình xây dựng công thức tựa Visual Basic. Ngoài ra, bạn cũng có thể tạo ra các Field công thức nhanh chóng dựa vào hàng trăm công thức thư viện sẵn có, hoặc bạn có thể tạo ra công thức tùy biến của riêng mình. Phần cuối chương chúng ta cũng đã học về lệnh SQL, biểu thức SQL dùng truy vấn dữ liệu và thực hiện công việc tính toán trên máy chủ. Tất cả những kỹ năng này sẽ làm nền tảng cho các chương sau, khi chúng ta bắt tay vào xây dựng các báo biểu thực sự phức tạp.

Chương 6:

XÂY DỰNG BÁO BIỂU PHỨC HỢP

- ❖ *Tổng quan.*
- ❖ *Cách hoạt động của báo biểu con (Subreports).*
- ❖ *Định dạng báo biểu con (Subreports).*
- ❖ *Cách hoạt động của báo biểu ma trận (Cross-Tabs).*
- ❖ *Định dạng báo biểu ma trận (Cross-Tabs).*
- ❖ *Phân tích dữ liệu báo biểu ma trận (Cross-Tabs).*
- ❖ *Tóm tắt.*

1. TỔNG QUAN

Trong những năm qua, các doanh nghiệp và người dùng đã nhìn thấy một sự bùng nổ số lượng lớn của dữ liệu cần phải phân tích và trình bày. Thường dữ liệu thu thập từ nhiều nguồn khác nhau như nhập liệu từ ứng dụng, do hệ thống tính toán phát sinh, trích rút từ cơ sở dữ liệu, thư điện tử, bảng tính. Thông thường trước đây mỗi nguồn dữ liệu sẽ được xử lý riêng để cho ra một báo biểu. Tuy nhiên, trong phiên bản mới nhất này Crystal Reports hỗ trợ cho bạn một chức năng mạnh mẽ khác là Subreport cho phép kết hợp nhiều báo cáo thành một báo cáo duy nhất.

Trong phần đầu chương, bạn sẽ học cách tạo các báo biểu con Subreport liên kết và không liên kết, cách định dạng và kỹ thuật quản lý sử dụng Subreport. Trong phần thứ hai của chương, bạn sẽ học cách tạo, định dạng và sửa đổi các báo biểu ma trận dạng Cross-tab, loại báo biểu này rất thường gặp trong phân tích và thống kê.

2. CÁCH HOẠT ĐỘNG CỦA BÁO BIỂU CON

Sẽ có lúc bạn có nhu cầu muốn trình bày thông tin từ hai nguồn dữ liệu khác nhau trong cùng một báo biểu nào đó. Nếu hai nguồn dữ liệu này có một khóa chung và tương thích về dữ liệu, bạn có thể tạo ra một báo biểu dễ dàng bằng cách kết hợp chung hai nguồn dữ liệu khác nhau. Ví dụ, nếu có một bảng cơ sở dữ liệu nhân viên chứa tên, mã số nhân

viên và một cơ sở dữ liệu bảng lương cũng lưu trữ mã số nhân viên, bạn có thể kết nối hai nguồn dữ liệu này với nhau một cách dễ dàng để tạo ra một báo biểu liệt kê thông tin từ cả hai nguồn dữ liệu.

Nhưng nếu cơ sở dữ liệu bảng lương không lưu trữ mã số nhân viên tương đồng với bảng lương thì sao? Ví dụ, bảng nhân viên lưu mã số nhân viên dạng chuỗi, còn bảng lương lại lưu mã nhân viên dạng số. Lúc này sử dụng tính năng Subreport là thuận tiện nhất. Bạn có thể tạo một báo biểu từ bảng nhân viên (Employee), sau đó chèn thêm một Subreport để hiển thị thông tin bảng lương từ nguồn dữ liệu khác (hình 6.1).

Employee Payroll Report

<u>First Name</u>	<u>Last Name</u>	<u>Position</u>	<u>Hire Date</u>	<u>Salary</u>
Nancy	Davolio	Sales Representative	03/29/1995	\$40,000.00

Payroll Details				
<u>PayPeriodStart</u>	<u>PayPeriodEnd</u>	<u>GrossPay</u>	<u>Tax</u>	<u>NetPay</u>
01/03/2004	01/17/2004	\$1,538.46	\$353.85	\$826.92
01/18/2004	02/01/2004	\$1,538.46	\$353.85	\$826.92
02/02/2004	02/16/2004	\$1,538.46	\$353.85	\$826.92
02/17/2004	03/02/2004	\$1,538.46	\$353.85	\$826.92
03/03/2004	03/17/2004	\$1,538.46	\$353.85	\$826.92
03/18/2004	04/01/2004	\$1,538.46	\$353.85	\$826.92
04/02/2004	04/16/2004	\$1,538.46	\$353.85	\$826.92
04/17/2004	05/01/2004	\$1,538.46	\$353.85	\$826.92
05/02/2004	05/16/2004	\$1,538.46	\$353.85	\$826.92
05/17/2004	05/31/2004	\$1,538.46	\$353.85	\$826.92
06/01/2004	06/15/2004	\$1,538.46	\$353.85	\$826.92
06/16/2004	06/30/2004	\$1,538.46	\$353.85	\$826.92
07/01/2004	07/15/2004	\$1,538.46	\$353.85	\$826.92
07/16/2004	07/30/2004	\$1,538.46	\$353.85	\$826.92
07/31/2004	08/14/2004	\$1,538.46	\$353.85	\$826.92

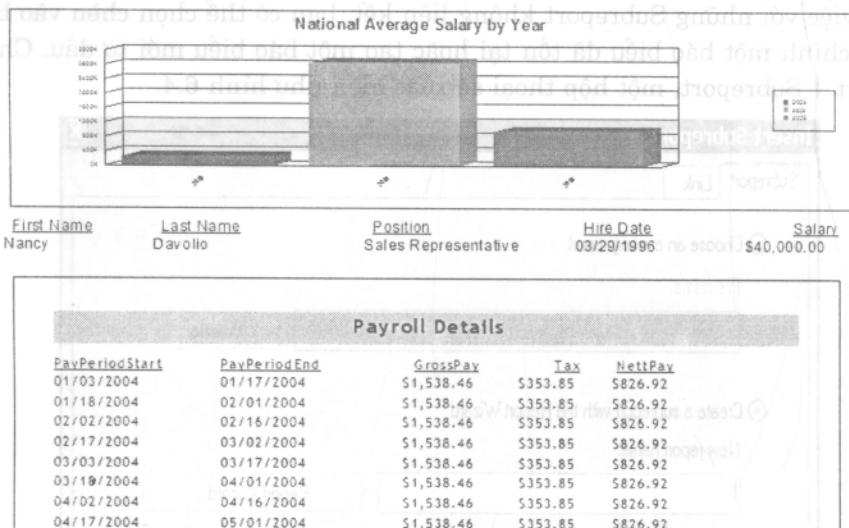
Hình 6.1: Báo biểu con Subreport được liên kết với báo biểu chính.

Subreport sử dụng hai khái niệm: Liên kết và không liên kết. Trong ví dụ vừa nêu, chúng ta sử dụng Subreport liên kết, như vậy mỗi nhân viên ở báo biểu chính sẽ liên kết với báo biểu con trình bày tiết bảng lương. Khi làm việc với các Subreport liên kết, Crystal Reports chuyển tham số giữa báo biểu chính và báo biểu con. Báo biểu con Subreport có thể sử dụng tham số cho việc lọc mẫu tin. Với Subreport không liên kết, Crystal Reports không thể chuyển bất kỳ tham số nào giữa báo biểu chính và báo biểu con.

Trong báo biểu ví dụ ở hình 6.2, bạn có thể thêm một báo biểu không liên kết vào trong báo biểu nhân viên để hiển thị mức lương trung

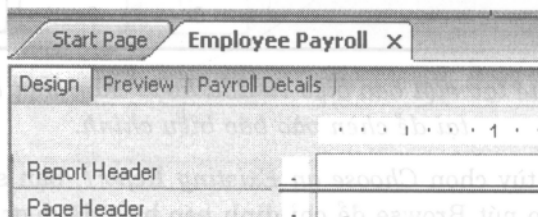
biểu ở dạng đồ thị. Do dữ liệu trong báo biểu chính và báo biểu con là không liên kết, bạn có thể thấy hai báo biểu hoàn toàn không liên quan gì với nhau (hình 6.2).

Employee Payroll Report



Hình 6.2: Báo biểu con không liên kết với báo biểu chính.

Khi làm việc với báo biểu con, một khái niệm quan trọng cần nhớ: báo biểu con là một phần tử của báo biểu chính. Mỗi báo biểu con chèn vào báo biểu chính như tất cả các thành phần báo biểu thông thường khác (hình 6.3).



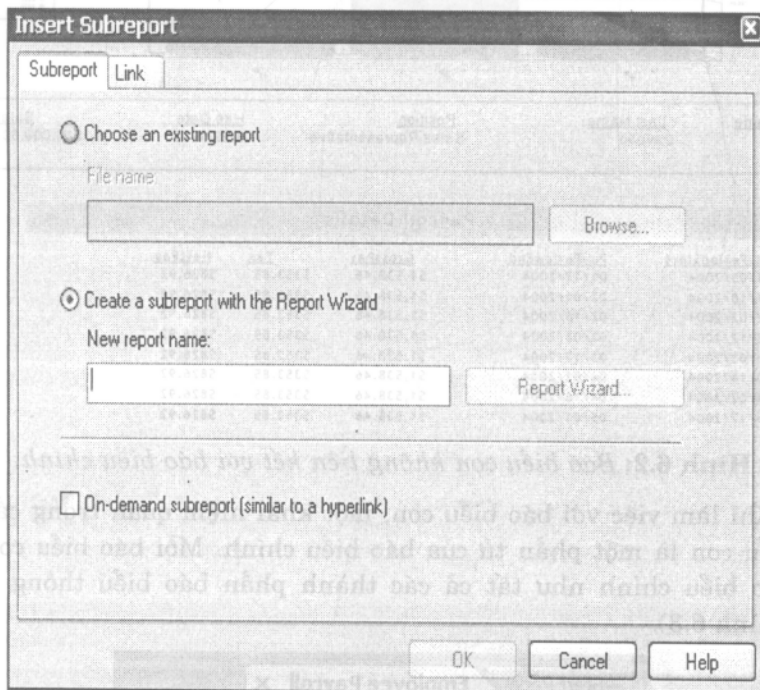
Hình 6.3: Báo biểu con trong tab Design.

Ghi chú: Bạn sẽ thấy một tab Preview cho mỗi báo biểu con.

Bạn có thể sửa đổi thuộc tính của đối tượng Subreport trong báo biểu chính hoặc trong bản thân báo biểu con Subreport. Để bắt đầu chúng ta sẽ học cách chèn một báo biểu con vào báo biểu chính.

2.1. Chèn Subreport không liên kết

Dạng báo biểu Subreport không liên kết không cần nhận tham số chuyển cho nó từ báo biểu chính và có thể trình bày bất kỳ thông tin nào mà bạn yêu cầu. Thông tin trình bày có thể không liên quan gì đến dữ liệu của báo biểu chính và có thể từ những nguồn cơ sở dữ liệu khác nhau. Khi làm việc với những Subreport không liên kết, bạn có thể chọn chèn vào báo biểu chính một báo biểu đã tồn tại hoặc tạo một báo biểu mới từ đầu. Chọn Insert | Subreport, một hộp thoại sẽ xuất hiện như hình 6.4.



Hình 6.4: Có thể tạo một báo biểu từ đầu hoặc chọn một báo biểu đã tồn tại để chèn vào báo biểu chính.

Nếu click tùy chọn *Choose an Existing Report*, bạn sẽ chọn báo biểu có sẵn. Click vào nút *Browse* để chỉ định báo biểu đã được tạo ra trước đó. Nếu click tùy chọn *Create a Subreport With the Report Wizard*, khi đó bạn phải nhập vào tên Subreport mới và sau đó click vào nút *Report Wizard* để tạo mới báo biểu. Khi đã hoàn tất việc tạo báo biểu mới hoặc chọn một báo biểu đã tồn tại, click *OK* để trở về hộp thoại *Insert Subreport*.

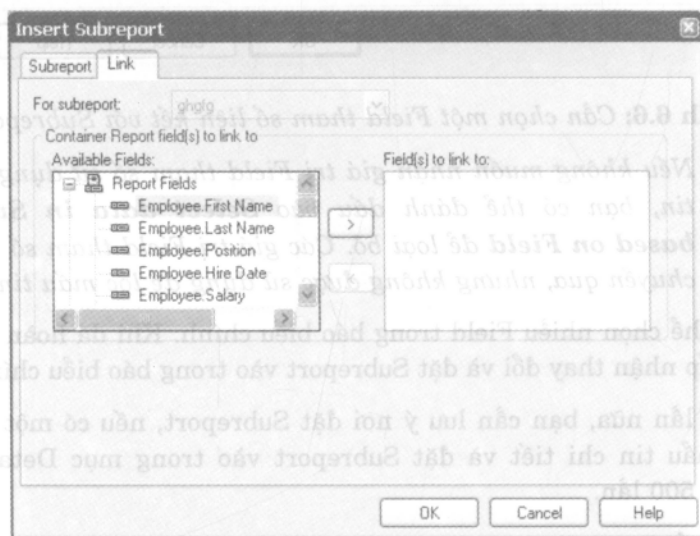
Khi đã hoàn tất với hộp thoại *Insert Subreport*, click *OK*. Một báo biểu con Subreport sẽ được gắn trên đỉnh của con trỏ chuột. Bạn di

chuyển chuột vào vùng báo biểu chính và click chuột vào vị trí đã xác định đặt báo biểu con.

Bạn cần phải chọn vị trí đặt Subreport cẩn thận vì nó ảnh hưởng đến kết quả xử lý của báo biểu con. Ví dụ, nếu đặt Subreport trong Report Header hoặc Report Footer, thì Subreport sẽ được xử lý chỉ một lần cho mỗi báo biểu. Nếu đặt Subreport trên Page Header hoặc Page Footer, thì Subreport sẽ được xử lý ở mọi trang.

2.2. Chèn một Subreport có liên kết

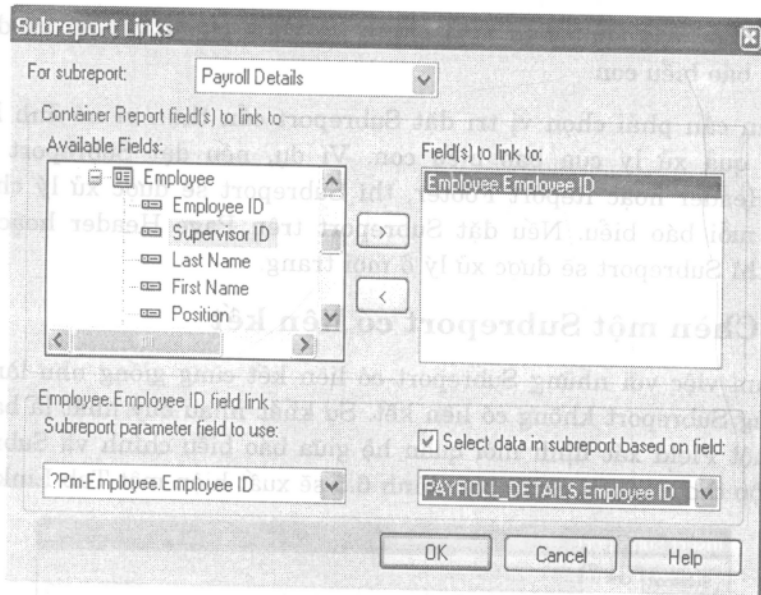
Làm việc với những Subreport có liên kết cũng giống như làm việc với những Subreport không có liên kết. Sự khác nhau duy nhất là bạn cần chỉ rõ một Field xác định mối quan hệ giữa báo biểu chính và Subreport. Trong hộp thoại Insert Subreport hình 6.5 sẽ xuất hiện một Tab Link mới.



Hình 6.5: Chỉ rõ một Field liên kết giữa báo biểu chính và Subreport.

Bước đầu tiên trong việc liên kết hai báo biểu là xác định một Field liên kết giữa báo biểu chính với Subreport. Từ danh sách các Field sẵn có bên trái trong hộp thoại, chọn một Field và click vào mũi tên để di chuyển Field từ danh sách sang hộp thoại bên phải.

Sau khi chọn Field, các tùy chọn bổ sung sẽ xuất hiện ở đáy của hộp thoại, như hình 6.6, cho phép chọn một Field trong Subreport. Chú ý rằng Field tham số được tạo tự động, điều bạn cần phải làm là lựa chọn một Field đang sử dụng trong Subreport.



Hình 6.6: Cần chọn một Field tham số liên kết với Subreport.

Ghi chú: Nếu không muốn nhận giá trị Field tham số sử dụng lọc mẫu tin, bạn có thể đánh dấu vào **Select data in Subreport based on Field** để loại bỏ. Các giá trị Field tham số vẫn được chuyển qua, nhưng không được sử dụng để lọc mẫu tin.

Có thể chọn nhiều Field trong báo biểu chính. Khi đã hoàn tất, click OK để chấp nhận thay đổi và đặt Subreport vào trong báo biểu chính.

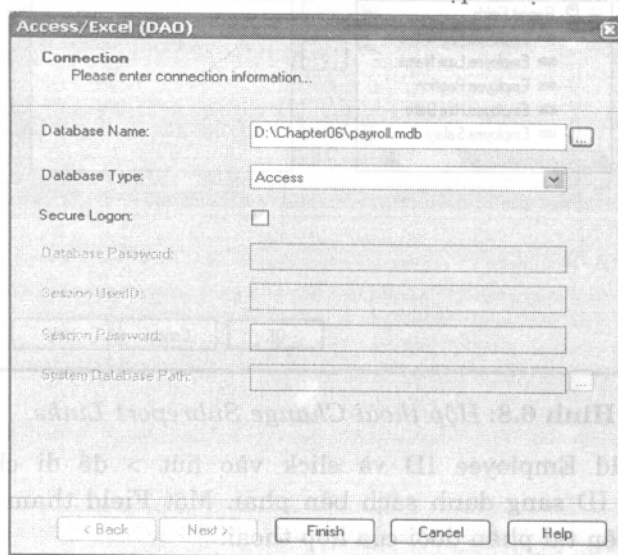
Một lần nữa, bạn cần lưu ý nơi đặt Subreport, nếu có một báo biểu với 500 mẫu tin chi tiết và đặt Subreport vào trong mục Detail, nó sẽ được xử lý 500 lần.

2.3. Kết hợp lại

Để sử dụng Subreport thành thạo, phần tiếp theo chúng ta sẽ xây dựng lại báo biểu Employee Payroll. Bắt đầu sẽ là một báo biểu đơn giản liệt kê danh sách nhân viên, sau đó ta sẽ thêm vào một Subreport để hiển thị chi tiết bảng lương từ một nguồn dữ liệu riêng biệt. Để tạo lại báo biểu này, thực hiện các bước sau:

1. Mở Crystal Reports và mở báo biểu Payroll.rpt từ đĩa CD ROM.
2. Chuyển sang tab Design để xem báo biểu và click Insert | Subreport từ menu để mở hộp thoại Insert Subreport.

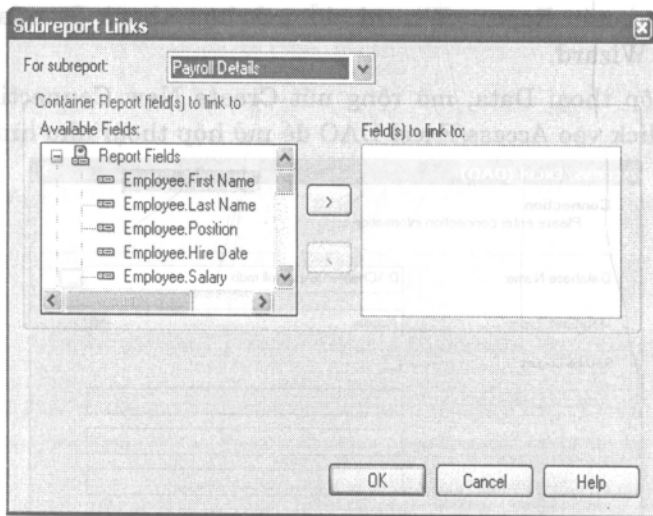
3. Click vào tùy chọn *Create a Subreport with the report Wizard*, sau đó nhập tên cho Subreport, trong ví dụ này ta đặt tên cho Subreport là Payroll Details.
4. Click vào nút Report Wizard để mở hộp thoại Standard Report creation Wizard.
5. Trong hộp thoại Data, mở rộng nút Create New Connection, sau đó double click vào Access/Excel DAO để mở hộp thoại như hình 6.7.



Hình 6.7: Tùy chọn dữ liệu kết nối Access / Excel.

6. Click vào nút (...), chọn file Payroll.mdb trong đĩa CD ROM, sau đó click nút Finish.
7. Mở rộng nút Tables nguồn dữ liệu, double click vào bảng Payroll_Details để di chuyển bảng Payroll_Details vào danh sách, sau đó click Next.
8. Trong hộp thoại Fields, click vào nút (>) để di chuyển tất cả các Field trong bảng Payroll_Details.
9. Click nút Finish để trở về hộp thoại Insert Subreport.
10. Click OK để hoàn tất việc tạo Subreport. Bây giờ Subreport sẽ được gắn vào đỉnh con trỏ chuột.
11. Đặt Subreport vào trong phần Details của báo biểu.
12. Chuyển sang chế độ Preview để kiểm tra kết quả kết xuất của Subreport là chính xác.

13. Tiếp theo, ta sẽ thiết lập liên kết cho Subreport. Click phải vào Subreport và chọn Change Subreport Links từ menu tắt để mở hộp thoại như hình 6.8.



Hình 6.8: Hộp thoại Change Subreport Links.

14. Chọn Field Employee ID và click vào nút > để di chuyển Field Employee ID sang danh sách bên phải. Một Field tham số được tạo và xuất hiện tại phần dưới của hộp thoại.
15. Phía bên phải của hộp thoại, click vào tùy chọn *Select data in Subreport based on Field* và sử dụng danh sách xổ xuống chọn Field Employee ID, sau đó click nút OK để trở về báo biểu chính.
16. Kết quả trong Subreport giờ đây sẽ lọc theo nhân viên.

Employee Payroll Report

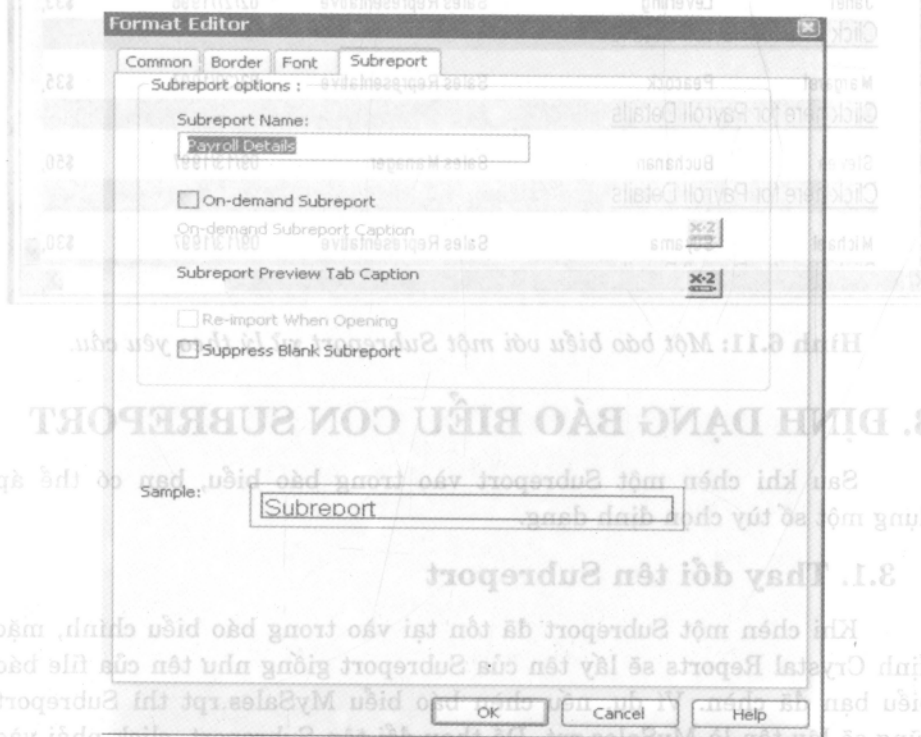
First Name	Last Name	Position	Hire Date	
Nancy	Davolio	Sales Representative	03/20/1998	\$40,000.00
5/13/2008				
PayPeriodStart	PayPeriodEnd	GrossPay	Tax	
10/21/2005 12:00:00AM	10/28/2005 12:00:00AM	\$712.82	\$156.92	
10/29/2005 12:00:00AM	11/4/2005 12:00:00AM	\$860.26	\$189.26	
11/4/2005 12:00:00AM	11/11/2005 12:00:00AM	\$637.63	\$140.32	
11/11/2005 12:00:00AM	11/18/2005 12:00:00AM	\$85.68	\$18.95	
11/18/2005 12:00:00AM	11/25/2005 12:00:00AM	\$510.59	\$112.33	
11/25/2005 12:00:00AM	12/2/2005 12:00:00AM	\$986.39	\$217.00	
12/2/2005 12:00:00AM	12/9/2005 12:00:00AM	\$480.04	\$105.61	
12/9/2005 12:00:00AM	12/16/2005 12:00:00AM	\$459.18	\$101.02	
12/16/2005 12:00:00AM	12/23/2005 12:00:00AM	\$327.50	\$72.05	
12/23/2005 12:00:00AM	12/30/2005 12:00:00AM	\$25.86	\$5.69	
12/30/2005 12:00:00AM	1/6/2006 12:00:00AM	\$996.09	\$219.14	

Hình 6.9: Báo biểu liên kết Subreport hiển thị chi tiết bảng lương.

2.4. Tạo một Subreport theo yêu cầu

Subreport theo yêu cầu (On-Demand) là những Subreport chỉ thực thi khi người sử dụng yêu cầu chúng. Để tạo một Subreport theo yêu cầu, bạn thực hiện các bước sau:

1. Mở Crystal Reports và mở báo biểu Payroll mà chúng ta đã làm việc trong phần trước.
2. Click phải vào Subreport, chọn Format Subreport từ menu tắt và click vào Tab Subreport để mở hộp thoại như hình 6.10.



Hình 6.10: Tùy chọn định dạng Subreport.

3. Chọn tùy chọn On Demand Subreport. Subreport sẽ được hiển thị như một liên kết trên báo biểu.
4. Để thông báo điều khiển xuất hiện trên báo biểu, click vào cả hai tùy chọn X+2, sau đó nhập chuỗi hướng dẫn (ví dụ, bạn nhập "Click here for Payroll Details").
5. Click OK để trở về báo biểu, bây giờ báo biểu sẽ trông như hình 6.1.

Employee Payroll Report				
First Name	Last Name	Position	Hire Date	
Nancy	Davolio	Sales Representative	03/29/1996	\$40,
Click here for Payroll Details				
Andrew	Fuller	Vice President, Sales	07/12/1996	\$90,
Click here for Payroll Details				
Janet	Leverling	Sales Representative	02/27/1996	\$33,
Click here for Payroll Details				
Margaret	Peacock	Sales Representative	03/30/1997	\$35,
Click here for Payroll Details				
Steven	Buchanan	Sales Manager	09/13/1997	\$50,
Click here for Payroll Details				
Michael	Suyama	Sales Representative	09/13/1997	\$30,

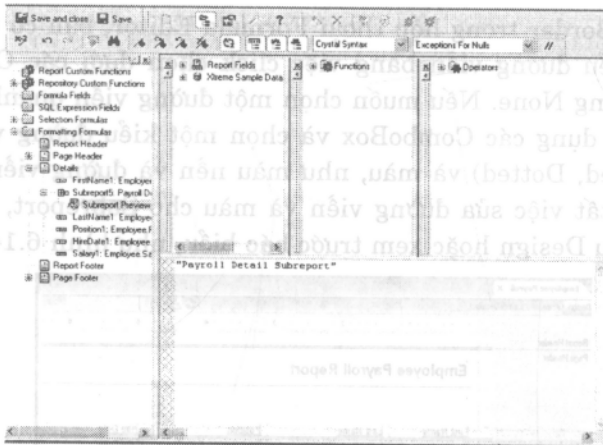
Hình 6.11: Một báo biểu với một Subreport xử lý theo yêu cầu.

3. ĐỊNH DẠNG BÁO BIỂU CON SUBREPORT

Sau khi chèn một Subreport vào trong báo biểu, bạn có thể áp dụng một số tùy chọn định dạng.

3.1. Thay đổi tên Subreport

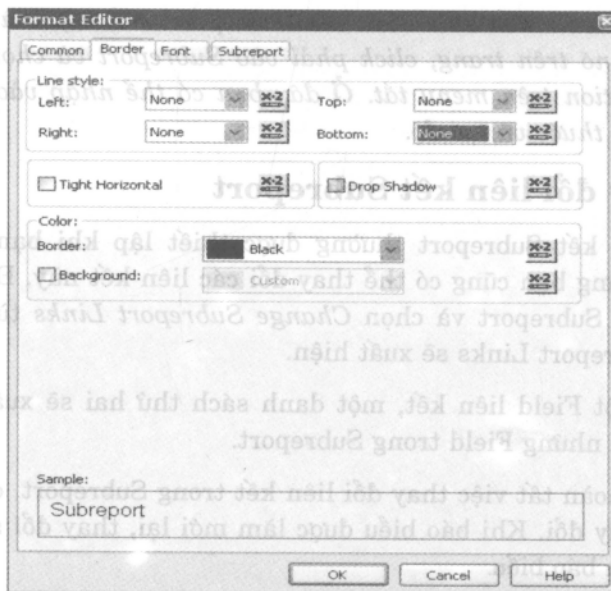
Khi chèn một Subreport đã tồn tại vào trong báo biểu chính, mặc định Crystal Reports sẽ lấy tên của Subreport giống như tên của file báo biểu bạn đã chèn. Ví dụ, nếu chèn báo biểu MySales.rpt thì Subreport cũng sẽ lấy tên là MySales.rpt. Để thay đổi tên Subreport, click phải vào Subreport và chọn Format Subreport từ menu tắt. Sử dụng tùy chọn trên trang thuộc tính Formula Editor, như hình 6.12, bạn có thể thay đổi tên của Subreport. Bất cứ khi nào tên báo biểu hiển thị (ví dụ, trong báo biểu chính, công cụ văn bản, trên tab Design cho Subreport v.v...) thì tên mới sẽ được sử dụng.



Hình 6.12: Có thể đổi tên Subreport sử dụng hộp thoại Formula Editor.

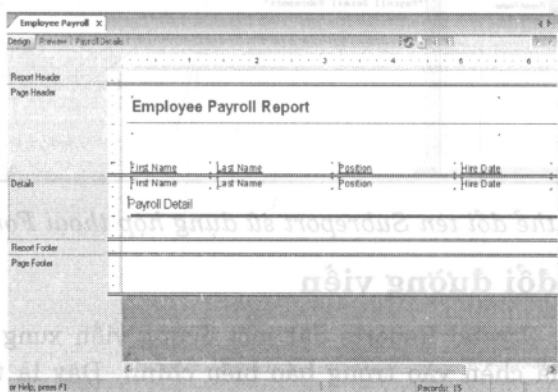
3.2. Thay đổi đường viền

Mặc định, Crystal Reports đặt một đường viền xung quanh bất kỳ Subreport nào đã chèn vào trong báo biểu chính. Đây là tùy chọn dạng mặc định thông thường. Để thay đổi đường viền quanh Subreport, click phải vào Subreport, chọn Formula Subreport và click vào tab Border để mở hộp thoại như hình 6.13.



Hình 6.13: Có thể định dạng Subreport từ menu tắt.

Từ tab Border trong hộp thoại Formula Editor, bạn có thể thay đổi tất cả bốn kiểu đường viền bằng việc click chọn dưới các ComboBox từ kiểu Single sang None. Nếu muốn chọn một đường viền quanh Subreport, bạn có thể sử dụng các ComboBox và chọn một kiểu đường viền (Single, Double, Dashed, Dotted) và màu, như màu nền và đường viền bóng dưới. Khi đã hoàn tất việc sửa đường viền và màu cho Subreport, click OK để trở về báo biểu Design hoặc xem trước báo biểu, như hình 6.14.



Hình 6.14: Di chuyển và định dạng đường viền xuất hiện quanh Subreport.

Ghi chú: Để điều khiển chính xác nơi Subreport xuất hiện và kích thước của nó trên trang, click phải vào Subreport và chọn *Size and Position* trên menu tắt. Ở đây bạn có thể nhập vào chính xác kích thước và tọa độ.

3.3. Thay đổi liên kết Subreport

Các liên kết Subreport thường được thiết lập khi bạn chèn một Subreport, nhưng bạn cũng có thể thay đổi các liên kết này. Để thay đổi, click phải vào Subreport và chọn *Change Subreport Links* từ menu tắt. Hộp thoại Subreport Links sẽ xuất hiện.

Chọn một Field liên kết, một danh sách thứ hai sẽ xuất hiện cho phép bạn chọn những Field trong Subreport.

Khi đã hoàn tất việc thay đổi liên kết trong Subreport, click OK để chấp nhận thay đổi. Khi báo biểu được làm mới lại, thay đổi sẽ được cập nhật ngay trên báo biểu.

Ghi chú: Bạn có thể làm mới lại báo biểu bất cứ lúc nào bằng việc click vào biểu tượng Refresh hoặc nhấn phím F5.

4. CÁCH HOẠT ĐỘNG CỦA BÁO BIỂU MA TRẬN (CROSS-TAB)

Cross-Tab là dạng báo biểu tổng kết phức hợp thường dùng để phân tích dữ liệu.

Trong chương này, bạn sẽ học cách tạo, định dạng và thao tác trên báo biểu Cross-Tab.

Báo biểu dạng Cross-Tab trông giống như hình 6.15, nó hình thành từ các hàng và cột tổng kết dữ liệu.

International and Domestic Sales

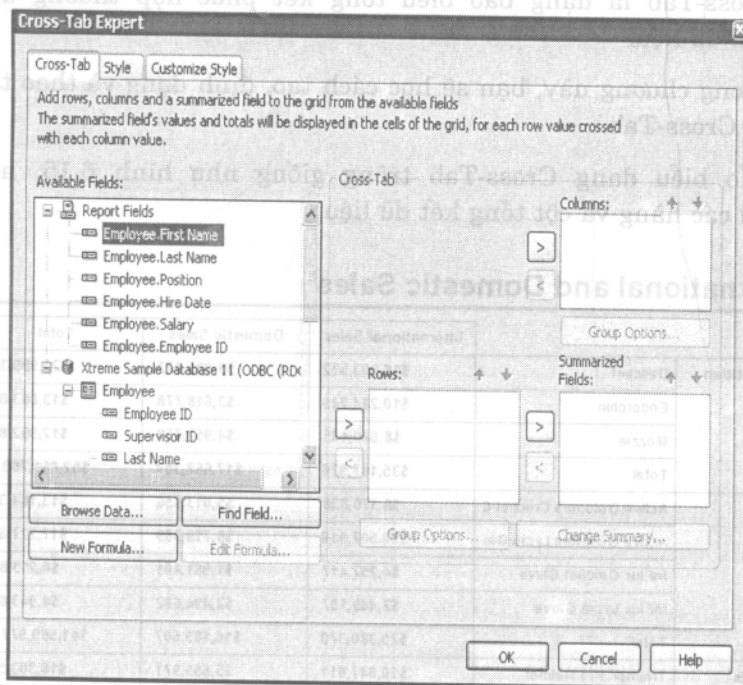
		International Sales	Domestic Sales	Total
Competition	Descent	\$16,903,532	\$9,095,823	\$25,999,355
	Endorphin	\$10,244,249	\$3,618,778	\$13,863,027
	Mozzie	\$8,040,145	\$4,952,752	\$12,992,897
	Total	\$35,187,926	\$17,667,354	\$52,855,280.12
Gloves	Active Outdoors Crochet C	\$8,170,838	\$5,013,556	\$13,184,394
	Active Outdoors Lycra Glo	\$10,507,918	\$6,719,969	\$17,227,887
	InFlux Crochet Glove	\$4,252,417	\$1,961,401	\$6,213,818
	InFlux Lycra Glove	\$2,449,197	\$2,494,682	\$4,943,879
	Total	\$25,380,370	\$16,189,607	\$41,569,977.61
Helmets	Triumph Pro Helmet	\$10,647,611	\$5,655,971	\$16,303,582
	Triumph Varigo Helmet	\$14,107,961	\$7,560,351	\$21,668,312
	Xtreme Adult Helmet	\$10,249,214	\$4,661,505	\$14,910,720
	Xtreme Youth Helmet	\$2,541,981	\$1,341,177	\$3,883,158
	Total	\$37,546,768	\$19,219,004	\$56,765,771.82

Hình 6.15: Báo biểu Cross-Tab thể hiện bảng tổng kết hàng bán trong và ngoài nước.

Báo biểu Cross-Tab nhìn tương tự như một bảng tính. Trong quá trình thiết kế báo biểu, Cross-Tab đơn giản chỉ hiển thị một dòng và cột. Khi xem báo biểu ở chế độ Preview, các dòng và các cột này sẽ chứa dữ liệu mà bạn yêu cầu. Kích thước của Cross-Tab được thay đổi dựa vào số mẫu tin trả về.

Để dễ dàng tạo những báo biểu Cross-Tab, Crystal Reports cung cấp công cụ Cross-Tab Expert giúp bạn tạo báo biểu Cross-Tab rất nhanh chóng. Ngoài ra, bạn cũng có thể chèn một báo biểu Cross-Tab vào báo biểu chính như những thành phần thông thường khác.

Từ menu Insert, click Insert + Cross-Tab. Một báo biểu Cross-Tab trống gắn liền với con trỏ chuột. Đặt Cross-Tab vào trong báo biểu chính và click phải vào Cross-Tab, chọn Cross-Tab Expert để mở hộp thoại như hình 6.16.

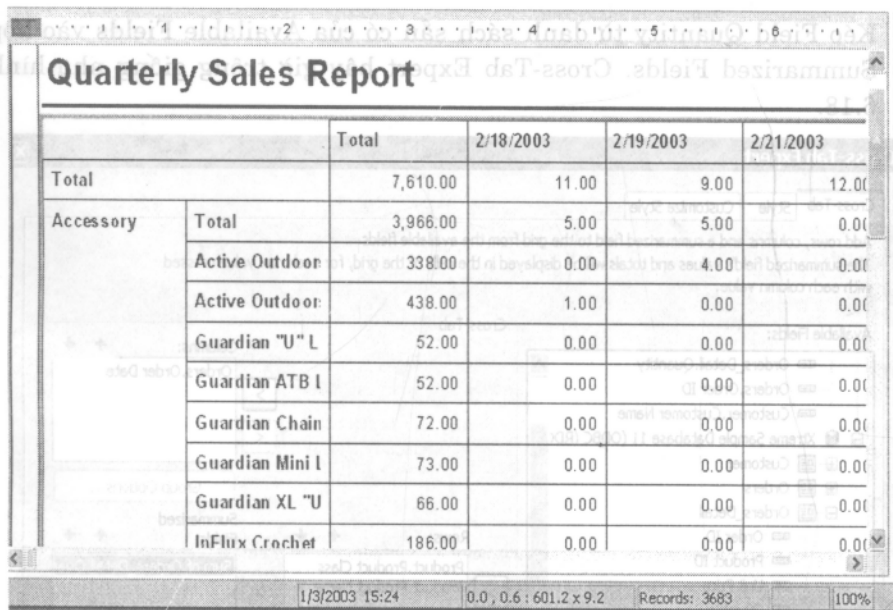


Hình 6.16: Hộp thoại Cross-Tab Expert.

Để tạo một Cross-Tab, bạn cần ba yếu tố: dòng, cột và các Field tính gộp. Với một Cross-Tab cơ bản, mỗi yếu tố bạn chỉ cần một Field. Để tạo mới các dòng, cột hoặc những Field tính gộp, bạn có thể kéo Field từ hộp thoại Available Fields tương ứng cho các dòng, các cột hoặc các Field tính gộp.

Với Cross-Tab, bạn có thể thêm nhiều Field cho các cột, các dòng và các Field tính gộp theo yêu cầu. Khi Cross-Tab in ra, các Field mà bạn chỉ định cho các cột và dòng sẽ được sử dụng để tạo ra một ma trận còn các Field tính gộp sẽ được tính toán tại điểm giao nhau của mỗi dòng và mỗi cột.

Ví dụ, một Cross-Tab như hình 6.17, Field Product Class và Product Name được sử dụng cho dòng và Field Order Date được sử dụng cho cột. Field Quantity sử dụng cho giá trị tổng gộp.



		Total	2/18/2003	2/19/2003	2/21/2003
Total		7,610.00	11.00	9.00	12.00
Accessory	Total	3,966.00	5.00	5.00	0.00
	Active Outdoor	338.00	0.00	4.00	0.00
	Active Outdoor:	438.00	1.00	0.00	0.00
	Guardian "U" L	52.00	0.00	0.00	0.00
	Guardian ATB I	52.00	0.00	0.00	0.00
	Guardian Chain	72.00	0.00	0.00	0.00
	Guardian Mini I	73.00	0.00	0.00	0.00
	Guardian XL "U	66.00	0.00	0.00	0.00
	InFlux Crochet	186.00	0.00	0.00	0.00

1/3/2003 15:24 0.0, 0.6 : 601.2 x 9.2 Records: 3683 100%

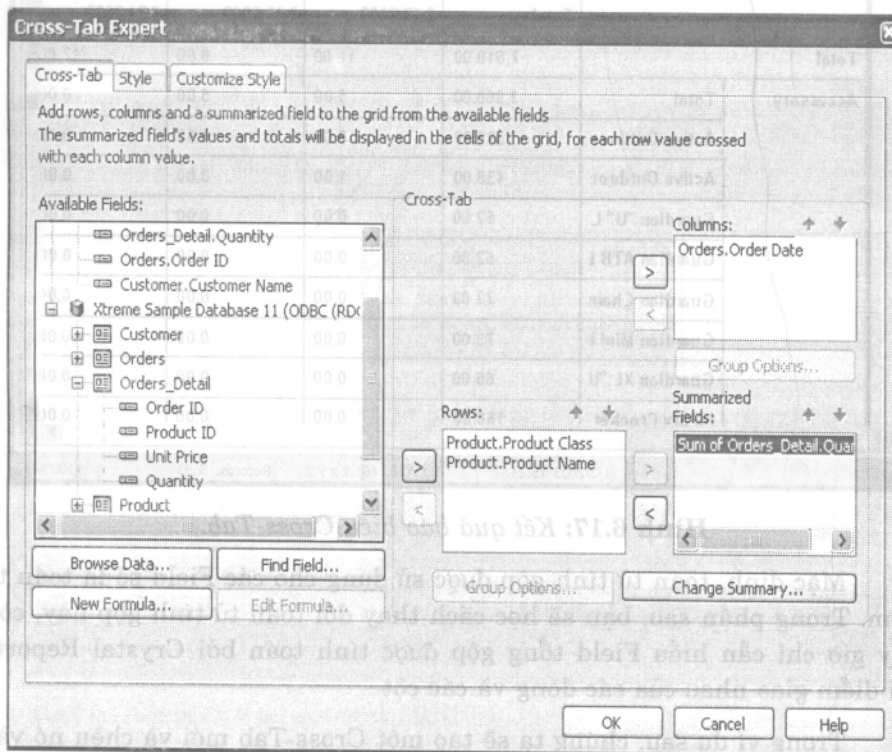
Hình 6.17: Kết quả báo biểu Cross-Tab.

Mặc định, toán tử tính gộp được sử dụng cho các Field số là toán tử Sum. Trong phần sau, bạn sẽ học cách thay đổi toán tử tính gộp này, còn bây giờ chỉ cần hiểu Field tổng gộp được tính toán bởi Crystal Reports tại điểm giao nhau của các dòng và các cột.

Trong ví dụ sau, chúng ta sẽ tạo một Cross-Tab mới và chèn nó vào trong một báo biểu hiện hành. Để làm được điều này, bạn thực hiện các bước sau:

1. Mở Crystal Reports và mở báo biểu Quarterlysales.rpt từ đĩa CD ROM.
2. Chuyển sang Tab Design của báo biểu và click Insert | Cross-Tabs. Sau đó đặt một Cross-Tab trống trong Report Header của báo biểu.
3. Click phải vào Cross-Tab và chọn Cross-Tab Expert từ menu tắt.
4. Kéo Field Product Class và Field Product Name từ danh sách sẵn có của Available Fields vào hộp Rows.
5. Kéo Field Order Data từ danh sách sẵn có của Available Fields vào hộp Columns.

- Kéo Field Quantity từ danh sách sẵn có của Available Fields vào hộp Summarized Fields. Cross-Tab Expert bây giờ trông giống như hình 6.18.

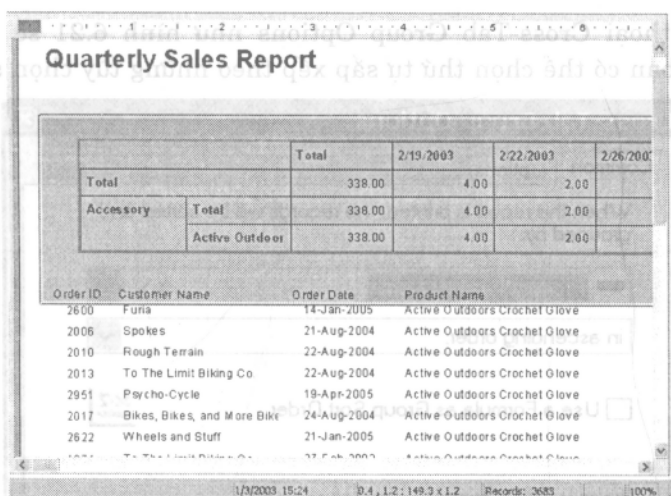


Hình 6.18: Cross-Tab Expert với tất cả các Field yêu cầu.

- Click OK để trở về báo biểu.

Khi xem báo biểu ở chế độ Preview, dữ liệu sẽ được chứa trong Cross-Tab và được hiển thị trong các dòng, các cột cùng các Filed tính tổng gộp mà chúng ta đã chọn.

Một điều quan trọng cần nhớ khi thêm một Cross-Tab vào trong báo biểu là Crystal Reports cho phép bạn chỉ chèn một Cross-Tab vào trong báo biểu, trong nhóm Header hay nhóm Footer. Nếu chèn Cross-Tab vào trong báo biểu Header hoặc Footer, Cross-Tab sẽ trình bày toàn bộ dữ liệu mà bạn đã yêu cầu, hiển thị tất cả các mẫu tin. Nếu đặt một Cross-Tab vào trong một nhóm Group Header hoặc Group Footer, Cross-Tab của bạn sẽ chỉ trình bày những mẫu tin liên kết trong một nhóm đặc biệt, như hình 6.19.



Quarterly Sales Report

		Total	2/19/2003	2/22/2003	2/26/2003
Total		338.00	4.00	2.00	
Accessory	Total	338.00	4.00	2.00	
	Active Outdoor	338.00	4.00	2.00	

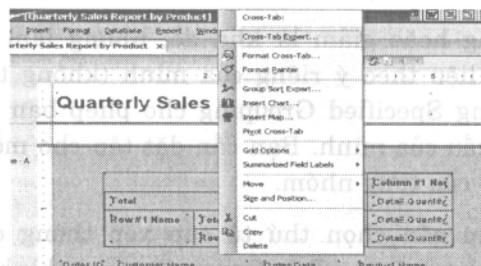
Order ID	Customer Name	Order Date	Product Name
2600	Furia	14-Jan-2005	Active Outdoors Crochet Glove
2006	Spokes	21-Aug-2004	Active Outdoors Crochet Glove
2010	Rough Terrain	22-Aug-2004	Active Outdoors Crochet Glove
2013	To The Limit Biking Co.	22-Aug-2004	Active Outdoors Crochet Glove
2951	Psycho-Cycle	19-Apr-2005	Active Outdoors Crochet Glove
2017	Bikes, Bikes, and More Bike	24-Aug-2004	Active Outdoors Crochet Glove
2622	Wheels and Stuff	21-Jan-2005	Active Outdoors Crochet Glove

Hình 6.19: Một Cross-Tab được đặt trong nhóm Group Header sẽ lọc theo nhóm.

Một trong những lỗi chung nhất khi đặt một Cross-Tab là việc đặt nó không đúng vị trí. Nếu bạn chèn một báo biểu trong Tab Preview, hãy chú ý con trỏ trên màn hình khi di chuyển ngang qua khu vực báo biểu. Nếu là biểu tượng cảm thì bạn không thể thêm Cross-Tab vào trong khu vực đó. Điều tốt nhất là đặt nó vào trong Report Header hoặc sử dụng tab Design để đặt Cross-Tab.

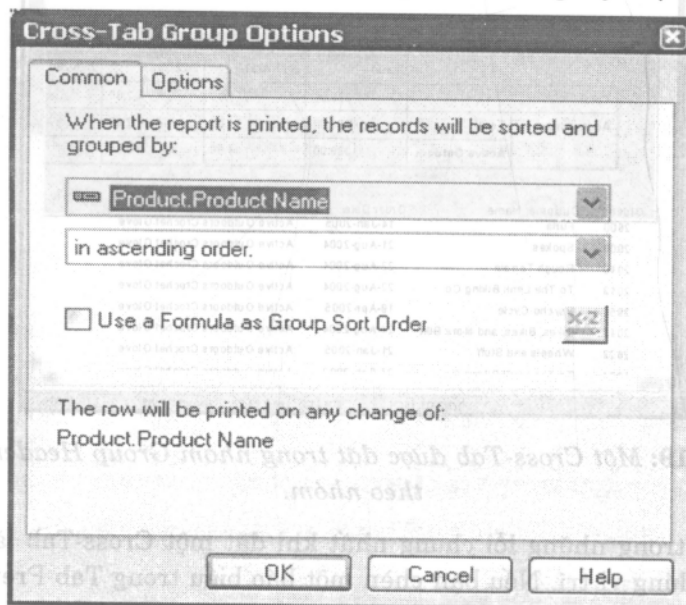
4.1. Tùy biến những tùy chọn nhóm (Group)

Bạn có thể điều khiển thông tin trình bày trong một Cross-Tab thông qua sử dụng nhóm (Group). Bạn có thể áp dụng khái niệm nhóm cho Cross-Tab bằng việc sử dụng menu tắt như hình 6.20 để hiệu chỉnh Cross-Tab, click vào nút Group Options.



Hình 6.20: Để sửa một Cross-Tab, chọn Cross-Tab, sau đó click phải vào nó và chọn Cross-Tab Expert.

Hộp thoại Cross-Tab Group Options như hình 6.21 sẽ được hiển thị. Ở đây bạn có thể chọn thứ tự sắp xếp theo những tùy chọn sau:



Hình 6.21: Một số tùy chọn nhóm sẵn có để sử dụng với Cross-Tab.

- Ascending (tăng dần).
- Descending (giảm dần).
- Specified Order (theo thứ tự đặc biệt).

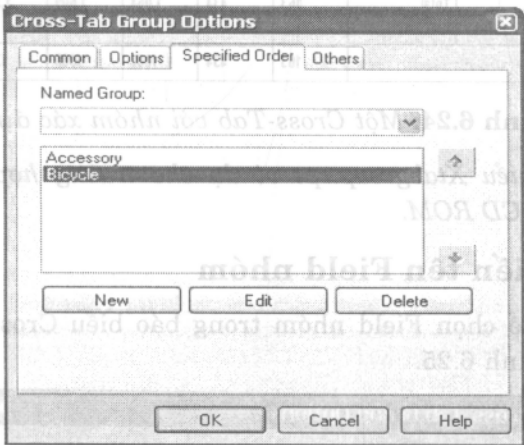
Khi sử dụng các tùy chọn sắp xếp thứ tự, nhớ rằng thứ tự sắp xếp này ảnh hưởng tới những mục trong Cross-Tab.

4.2. Sắp xếp dữ liệu nhóm

Sắp xếp tăng hoặc giảm là khá dễ, nhưng làm thế nào khi bạn muốn sắp xếp dữ liệu theo ý riêng của mình (không theo thứ tự thông thường)? Chức năng Specified Grouping cho phép bạn tạo ra một nhóm sắp xếp theo yêu cầu của mình. Bạn cần đặt tên cho mỗi nhóm và chỉ rõ tiêu chuẩn sắp xếp của từng nhóm.

Đầu tiên bạn cần chọn thứ tự sắp xếp thông qua tab Specified Order trong hộp thoại Cross-Tab Group Options (hình 6.22). Bước tiếp theo là định nghĩa tên nhóm, sau đó chỉ rõ tiêu chuẩn sắp xếp nhóm. Sau khi tất cả các tên nhóm được định nghĩa, bạn chọn từng nhóm và click

nút Edit để chỉ rõ tiêu chuẩn sắp xếp. Để thiết lập tiêu chuẩn cho lọc mẫu tin cho nhóm, sử dụng menu xổ xuống để chọn toán tử và giá trị. Mặc định, tất cả các mẫu tin còn lại (không thỏa mãn tiêu thức lọc của các nhóm) sẽ được đặt trong một nhóm mang tên Others, như hình 6.23. Bạn cũng có thể chọn để loại bỏ tất cả các mẫu tin không thỏa tiêu thức lọc của nhóm.



Hình 6.22: Tab Specified Order.

Sau khi đã định nghĩa các nhóm và xác định tiêu thức, click OK để chấp nhận thay đổi các tùy chọn nhóm.



Hình 6.23: Các tùy chọn dùng chỉ định nhóm khi tiêu thức lọc mẫu tin không thỏa.

Kết quả dữ liệu nhóm sẽ sắp xếp lại trong Cross-Tab như hình 6.24.

		QTR1	QTR4	QTR1	QTR2	QTR3
BoysBikes	Accessory	68.00	252.00	508.00	878.00	664.00
	Bicycle	95.00	182.00	556.00	727.00	670.00
	Total	163	434	1,064	1,605	1,334
Total		163	434	1064	1605	1334

Hình 6.24: Một Cross-Tab với nhóm xác định.

Ghi chú: Báo biểu Xtabgroup.rpt ví dụ cho trường hợp này được chứa trong CD ROM.

4.3. Tùy biến tên Field nhóm

Bạn có thể chọn Field nhóm trong báo biểu Cross-Tab thông qua hộp thoại như hình 6.25.



Hình 6.25: Có thể đặt vài tùy chọn nhóm sẵn có cho những nhóm được chèn vào trong báo biểu.

Tên Field nhóm có thể chọn theo Field trong cơ sở dữ liệu hoặc Field công thức. Nếu trong cơ sở dữ liệu có một bảng chứa tên mà bạn muốn sử dụng, bạn có thể chỉ rõ tên Field.

Ví dụ, nếu báo biểu Cross-Tab sử dụng Field Product ID, bạn có thể tùy biến chọn tên nhóm và chọn Field Product Name, như vậy tên sản phẩm sẽ xuất hiện thay cho Product ID. Nếu tên của nhóm không được lưu trữ trong bảng cơ sở dữ liệu, bạn có thể tạo ra một công thức để sử dụng như một tên nhóm.

Giải pháp tốt hơn là sử dụng công thức làm tên nhóm. Để tạo một công thức cho mục đích này, click vào nút X+2 bên cạnh *Use a Formula as Group Name*. Hộp thoại Formula Workshop sẽ hiển thị, cho phép bạn nhập vào một công thức làm tên nhóm hiển thị trong báo biểu.

Ví dụ chúng ta muốn hiển thị quý dựa vào tháng của Field Order Date. Như vậy công thức sẽ như sau:

```
If month ({Purchases.Order Date}) in [1,2,3] then "QTR1"  
else
```

```
If month ({Purchases.Order Date}) in [4,5,6] then "QTR2"  
else
```

```
If month ({Purchases.Order Date}) in [7,8,9] then "QTR3"  
else
```

```
If month ({Purchases.Order Date}) in [10,11,12] then  
"QTR4"
```

Bạn có thể tạo bất kỳ công thức nào mình thích, chỉ cần nó được thiết kế để trả về một chuỗi dùng làm tên hiển thị cho nhóm.

4.4. Làm việc với các Field tính gộp

Các Field tính gộp được sử dụng để tổng kết thông tin chứa trong các ô của báo biểu Cross-Tab. Mặc định, Crystal Reports sử dụng toán tử Sum cho các Field số, bạn có thể chèn vào trong các Field tính gộp toán tử Count để thực hiện phép đếm số lần xuất hiện hay các phép toán tính Max, Min và trung bình cộng. Để thay đổi kiểu công thức tính gộp, bạn chọn Field và click vào nút **Change Summary**. Các công thức tính có thể thay đổi là:

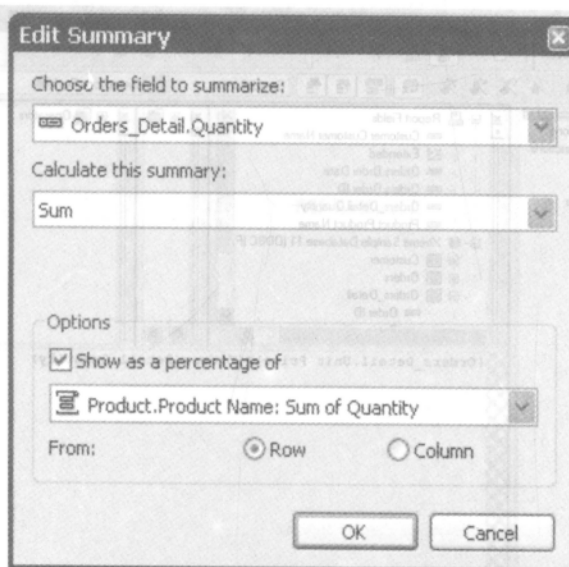
- ◆ Sum
- ◆ Average
- ◆ Minimum
- ◆ Maximum

- ♦ Count
- ♦ Discount
- ♦ Correlation
- ♦ Covariance
- ♦ Weighted average
- ♦ Median
- ♦ Pth percentile
- ♦ Nth largest
- ♦ Smallest
- ♦ Mode
- ♦ Nth most frequent
- ♦ Variance
- ♦ Standard Deviation
- ♦ Population Variance
- ♦ Population Standard Deviation
- ♦ NthSmallest

Ngoài những phép tính gộp chuẩn này, bạn còn có thể áp dụng tính theo tỉ lệ phần trăm như:

- ♦ Percent of Summary
- ♦ Percent of Average
- ♦ Percent of Maximum
- ♦ Percent of Minimum
- ♦ Percent of Count
- ♦ Percent of Distinct Count

Một vài hàm có thể yêu cầu bạn chỉ rõ Field hoặc một giá trị bổ sung. Hình 6.26 là một Field tổng gộp có thêm phép tính phần trăm dựa trên giá trị của toàn bộ cột tổng.



Hình 6.26: Field tổng gộp cần thêm tham số.

4.5. Chèn các Field công thức

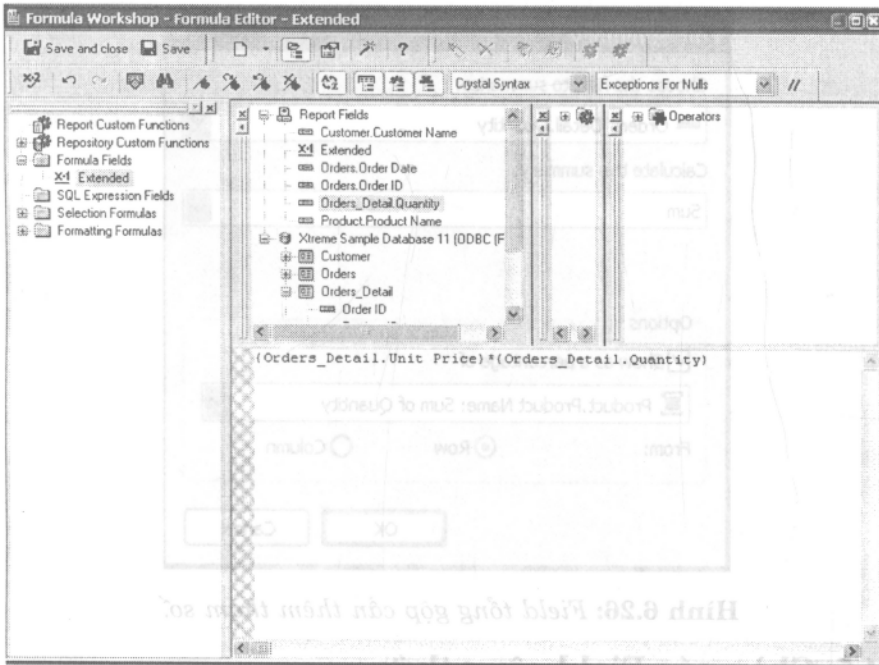
Ngoài những Field trong cơ sở dữ liệu, bạn còn có thể sử dụng các Field công thức trong Cross-Tab. Để chèn một Field công thức vào trong báo biểu Cross-Tab, click vào nút New Formula bên phải của hộp thoại Format Cross-Tab.

Ghi chú: Để hiệu chỉnh một Cross-Tab đã tồn tại, click phải vào đối tượng Cross-Tab và chọn Format Cross-Tab từ menu tắt.

Như thường lệ, đầu tiên bạn cần nhập vào một tên cho công thức muốn tạo. Click OK để mở hộp thoại Crystal Reports Formula Editor và nhập vào đoạn công thức, như hình 6.27.

Sử dụng hộp thoại Formula Editor, nhập vào đoạn công thức. Khi đã hoàn tất việc sửa công thức, click nút Save, sau đó click nút Close để đóng công thức và trở về hộp thoại Format Cross-Tab.

Công thức bây giờ sẽ xuất hiện trong danh sách Available Fields. Bạn có thể kéo công thức vào khung Row, Column hoặc Summarized Field để đưa nó vào báo biểu Cross-Tab. Khi báo biểu xem ở chế độ Preview, công thức này sẽ được ước lượng và hiển thị kết quả như hình 6.28.



Hình 6.27: Tạo công thức sử dụng Cross-Tab Expert.

Quarterly Sales Report

		QTR1	QTR4	QTR1
Accessory	Active Outdoors	11	25	27
	Crochet Glove	\$159.50	\$356.70	\$387.50
	Active Outdoors			
	Lycra Glove	\$64.35	\$610.50	\$958.50
	Guardian "U" Lock	\$0.00	\$35.00	\$85.76

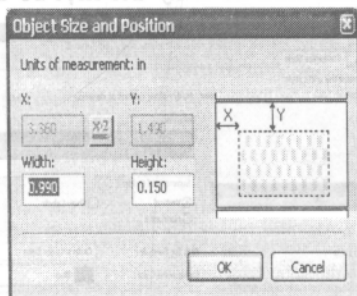
Hình 6.28: Ví dụ một công thức được chèn vào trong một Cross-Tab.

5. ĐỊNH DẠNG CROSS-TAB

Bạn đã biết cách chèn một đối tượng Cross-Tab vào trong báo biểu và điều khiển cấu trúc của nó, bây giờ chúng ta sẽ xem cách định dạng một đối tượng Cross-Tab.

5.1. Thay đổi Field, cột, dòng và kích thước

Bạn có thể thay đổi kích thước của một Field bằng việc click vào Field để chọn nó, sau đó sử dụng con trỏ thay đổi độ rộng và độ cao của Field. Bạn cũng có thể click phải vào Field và chọn **Size and Position** từ menu tắt để mở hộp thoại như hình 6.29, nhập kích thước chính xác của các Field thông qua các ô điều khiển trên hộp thoại.



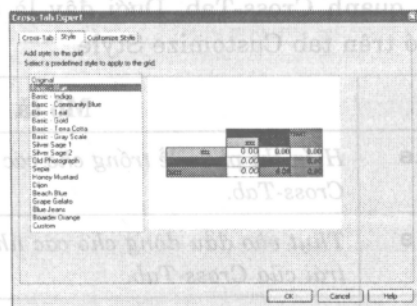
Hình 6.29: Hộp thoại đối tượng Size and Position.

Ghi chú: Khi hoàn tất việc thay đổi kích thước các Field, các cột hoặc các dòng, click vào bất cứ nơi nào ở ngoài Cross-Tab để trở về chế độ thiết kế đang làm việc.

5.2. Sử dụng các kiểu định dạng có sẵn

Do phần lớn thời gian làm việc với Cross-Tab là định dạng chúng nên để đỡ tốn thời gian, Crystal Reports cung cấp một số kiểu định dạng sẵn bạn có thể sử dụng ngay.

Để sử dụng một kiểu định dạng có sẵn cho đối tượng Cross-Tab, trong hộp thoại Cross-Tab, click vào Tab Style như hình 6.30.

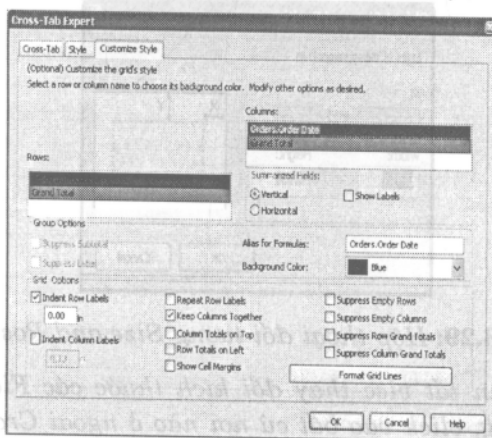


Hình 6.30: Sử dụng một kiểu được tạo trước đến Cross-Tab hoặc lưới OLAP.

Để thấy trước các kiểu đang định dạng, click vào kiểu định dạng trong danh sách bên trái. Ô vuông trên cạnh phải của hộp thoại sẽ hiển thị hình ảnh Cross-Tab mẫu để bạn hình dung kết quả.

5.3. Các kiểu định dạng tùy biến

Bạn có thể sử dụng tùy biến trong tab Customize Style của hộp thoại Cross-Tab Expert như hình 6.31 để thay đổi một số định dạng đặc biệt.



Hình 6.31: Có thể thêm tùy biến một kiểu được tạo trước sử dụng trong hộp thoại Cross-Tab Expert.

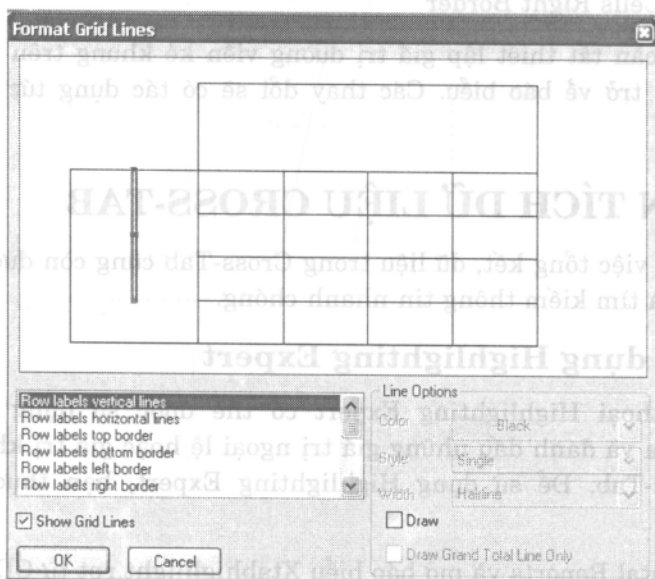
Để thay đổi định dạng đặc biệt của dòng và cột, chọn dòng hoặc cột từ danh sách có sẵn, sau đó sử dụng danh sách xổ xuống bên dưới để thay đổi màu nền và thiết lập những tùy chọn khác, như hiển thị tên nhân, đặt bí danh cho công thức v.v...

Phần dưới của hộp thoại là những tùy chọn để định dạng cho khung lưới xuất hiện quanh Cross-Tab. Dưới đây là các tùy chọn đường viền khung lưới sẵn có trên tab Customize Style.

Tùy chọn	Mô tả
Show Cell Margins	Hiển thị phần lề trống cho các ô bên trong Cross-Tab.
Indent Row Labels	Thụt vào đầu dòng cho các nhãn xuất hiện bên trái của Cross-Tab.
Repeat Row Labels	Lặp lại tên nhân của dòng trên tất cả các trang mới.

Keep Columns Together	Giữ tất cả các cột nằm chung nhau trên cùng trang.
Row Totals on Top	Di chuyển dòng tổng số từ vị trí mặc định của nó ở dưới Cross-Tab lên trên đầu.
Column Total on Left	Di chuyển cột tổng số từ vị trí mặc định của nó sang cạnh trái của Cross-Tab.
Suppress Empty Rows	Loại bỏ tất cả các dòng trống trong Cross-Tab.
Suppress Empty Columns	Loại bỏ tất cả các cột trống trong Cross-Tab.
Suppress Row Grand Totals	Ngăn không cho dòng tổng cộng xuất hiện theo mặc định ở cuối Cross-Tab.
Suppress Column Grand Totals	Ngăn cột tổng cộng xuất hiện theo mặc định trên cạnh phải của Cross-Tab.

Bạn cũng có thể điều khiển đường viền trên khung lưới xuất hiện trong Cross-Tab bằng cách click vào nút Format Grid Lines để mở hộp thoại như hình 6.32.



Hình 6.32: Bạn cũng có thể định dạng những hàng riêng lẻ trên lưới.

Sử dụng hộp thoại *Format Grid Lines*, chọn mục Cross-Tab, sau đó chọn lựa màu, kiểu và độ rộng cho đường viền theo 4 cạnh:

- ◆ Row Labels Vertical Lines
- ◆ Row Labels Horizontal Lines
- ◆ Row Labels Top Border
- ◆ Row Labels Bottom Border
- ◆ Row Labels Right Border
- ◆ Column Labels Vertical Lines
- ◆ Column Labels Horizontal Lines
- ◆ Column Labels Top Border
- ◆ Column Labels Bottom Border
- ◆ Column Labels Right Border
- ◆ Cells Vertical Lines
- ◆ Cells Horizontal Lines
- ◆ Cells Bottom Border
- ◆ Cells Right Border

Khi hoàn tất thiết lập giá trị đường viền kẻ khung trên Cross-Tab, click OK để trở về báo biểu. Các thay đổi sẽ có tác dụng tức thời trong Cross-Tab.

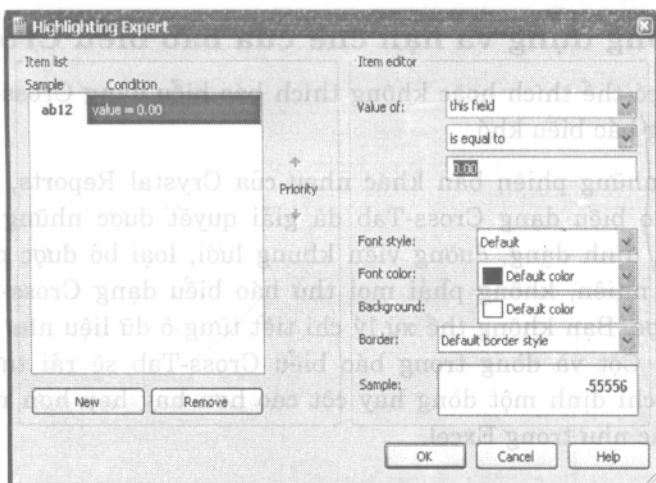
6. PHÂN TÍCH DỮ LIỆU CROSS-TAB

Ngoài việc tổng kết, dữ liệu trong Cross-Tab cũng còn được dùng để phân tích và tìm kiếm thông tin nhanh chóng.

6.1. Sử dụng Highlighting Expert

Hộp thoại Highlighting Expert có thể được sử dụng để nhanh chóng tìm ra và đánh dấu những giá trị ngoại lệ hoặc giá trị khác thường trong Cross-Tab. Để sử dụng Highlighting Expert, bạn thực hiện các bước sau:

1. Mở Crystal Reports và mở báo biểu Xtabhighlight.rpt từ CD ROM.
2. Click phải vào các Field trong Cross-Tab bạn muốn xác định thông tin tìm kiếm. Trong ví dụ này, bạn click phải vào Field Quantity. Từ menu tắt, chọn Highlighting Expert để mở hộp thoại như hình 6.33.



Hình 6.33: Hộp thoại Highlighting Expert.

3. Click vào nút New trên cạnh trái của hộp thoại.
4. Tiếp theo, trong mục Item Editor trên cạnh phải của hộp thoại, chọn toán tử Value Is từ danh sách sẵn có. Trong ô ngay bên dưới toán tử, nhập các tiêu thức tìm kiếm. Ví dụ, trong báo biểu này, chọn toán tử Is less than và nhập 10000.
5. Bây giờ chọn phong màu, màu nền và đường viền sẽ áp dụng khi tiêu thức này thỏa. Trong báo biểu này, chọn màu nền là Yellow. Nếu muốn nhập nhiều tiêu thức, click New trên Item List bên cạnh trái.

Ghi chú: Để thay đổi thứ tự ưu tiên của tiêu thức tìm kiếm, bạn sử dụng mũi tên lên và xuống.

6. Sau khi đã nhập các tiêu thức và tùy chọn định dạng, click OK để thoát khỏi hộp thoại Highlighting Expert.

Khi xem báo biểu ở chế độ Preview, những Field mà trước đó bạn lựa chọn trong Cross-Tab nếu có giá trị thỏa mãn tiêu thức sẽ được định dạng như bạn vừa chỉ định trong Highlighting Expert.

6.2. Thay đổi hướng Cross-Tab

Để làm cho Cross-Tab trở nên dễ đọc hơn, bạn có thể xoay Cross-Tab bằng cách click phải vào Cross-Tab và chọn Pivot Cross-Tab từ menu tắt. Mục chọn này sẽ hoán chuyển dòng và cột của Cross-Tab với nhau.

6.3. Công dụng và hạn chế của báo biểu Cross-Tab

Bạn có thể thích hoặc không thích báo biểu dạng Cross-Tab vì đây là một dạng báo biểu khó.

Qua những phiên bản khác nhau của Crystal Reports, cùng nhiều cải tiến báo biểu dạng Cross-Tab đã giải quyết được những vấn đề về ngắt trang, định dạng, đường viền khung lưới, loại bỏ được dòng và cột trống. Tuy nhiên, không phải mọi thứ báo biểu dạng Cross-Tab đều có thể làm được. Bạn không thể xử lý chi tiết từng ô dữ liệu như trong bảng tính Excel. Cột và dòng trong báo biểu Cross-Tab sẽ rải tự động, bạn không thể chỉ định một dòng hay cột cao hơn hay hẹp hơn những dòng hay cột khác như trong Excel.

7. TÓM TẮT

Báo biểu con hay còn gọi là Subreport giúp bạn tạo một báo biểu phức hợp, dữ liệu có thể tổng hợp từ hai hoặc nhiều nguồn dữ liệu khác nhau do nhiều báo biểu con tạo thành. Báo biểu con có thể liên kết hoặc không liên kết với báo biểu chính. Báo biểu ma trận Cross-Tab là một dạng báo biểu con mà bạn có thể chèn vào báo biểu chính. Báo biểu Cross-Tab trình bày dữ liệu tổng kết theo dòng và cột, thông tin trình bày của báo biểu dạng này thường cô đọng, ngắn gọn và rất dễ phân tích. Tuy nhiên, để phân tích báo biểu, Crystal còn cung cấp cho bạn một công cụ mạnh mẽ khác cũng rất hiệu quả đó là biểu đồ. Chương tiếp theo chúng ta sẽ học cách vẽ biểu đồ hay đồ thị trong báo biểu.

Chương 7:

BIỂU ĐỒ VÀ CÁC THÀNH PHẦN TRỰC QUAN

- ❖ *Tổng quan.*
- ❖ *Thêm biểu đồ vào báo biểu.*
- ❖ *Định dạng nâng cao.*
- ❖ *Tóm tắt.*

1. TỔNG QUAN

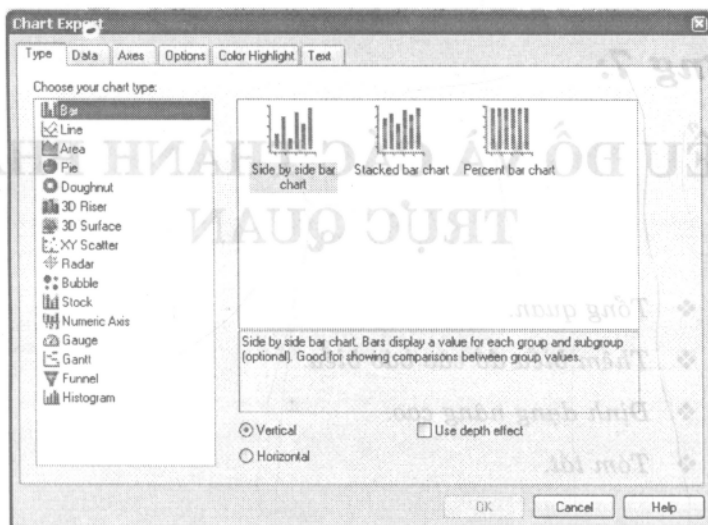
Dữ liệu ngày càng bùng nổ trong các tổ chức, công ty, xí nghiệp. Phần lớn những ứng dụng thương mại và các hệ thống thông tin đều liên quan đến cơ sở dữ liệu (CSDL). Một tổ chức có thể có thu thập hàng triệu mẫu tin để đánh giá phân tích thống kê. Crystal Reports là một giải pháp tất yếu để giải quyết vấn đề này, mỗi một báo biểu có thể được hình thành từ hàng triệu dòng thông tin theo một khuôn dạng gọn gàng và sáng sủa.

Bạn đã xem qua một số cách mà Crystal Reports thể hiện thông tin trên báo biểu, như tổng gộp số liệu, báo biểu phân cấp, tổng chạy, báo biểu ma trận Cross-Tab. Tất cả những hình thức trình bày dữ liệu này đều dưới dạng văn bản và con số.

Trong chương này, chúng ta sẽ khảo sát cách trình bày dữ liệu trực quan khác dưới dạng trực quan bằng hình ảnh thông qua biểu đồ và đồ thị. Với biểu đồ, bạn có thể đúc kết hàng triệu dòng dữ liệu vào trong một hình mô tả dễ nhận biết và phân tích hơn (hình 7.1).

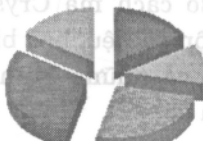
Bộ biểu đồ mà Crystal cung cấp có thể phục vụ cho đủ mọi lĩnh vực thương mại, khoa học, nghiên cứu, phân tích với hơn 40 mẫu dựng sẵn để bạn chọn lựa. Những biểu đồ này có thể được tùy biến, chỉnh sửa theo ý bạn để có thể đạt được kết quả như mong muốn.

Biết cách sử dụng biểu đồ, báo biểu của bạn có thể truyền đạt thông tin cô đọng và ấn tượng về dữ liệu như trong hình 7.2.



Hình 7.1: Các dạng biểu đồ Crystal Reports hỗ trợ.

Sales by Region



■ Berkshire
■ Alsace
■ Bayern
■ Lombardie
■ P.A.

■ Greater London
■ Aquitaine
■ Berlin
■ Piedmonte
■ WI

■ Hampshire
■ Ile de France
■ Hamburg
■ C.A.

■ Nottinghamshire
■ Languebec-Roussillon
■ Nordrhein-Westfalen
■ IC

■ West Yorkshire
■ Rhône-Alpes
■ Lazio
■ HA

Hình 7.2: Các biểu đồ gắn vào báo biểu.

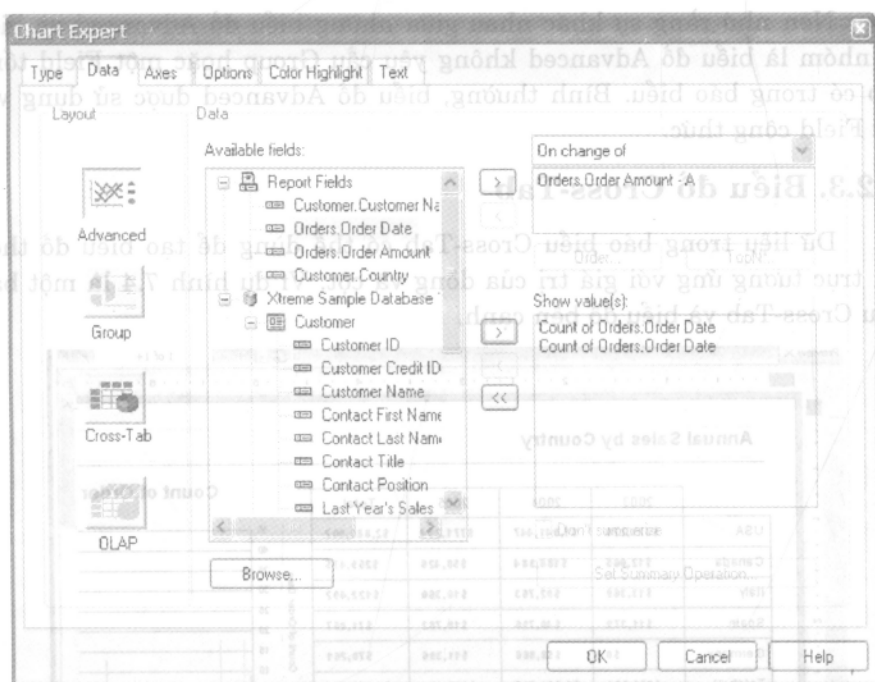
Trong chương này, chúng ta sẽ xem xét chi tiết từng biểu đồ, bắt đầu từ các biểu đồ thư viện mà Crystal cung cấp sẵn.

Các báo biểu tạo trong chương này chứa trong thư mục Project trên đĩa CD ROM. Bạn có thể thực hành các bước theo chỉ dẫn tạo báo biểu

trong giáo trình hoặc có thể chép báo biểu từ CD ROM vào trong ổ đĩa cứng của bạn rồi sửa lại.

2. THÊM BIỂU ĐỒ VÀO BÁO BIỂU

Crystal Reports hỗ trợ nhiều dạng biểu đồ khác nhau như biểu đồ cột, biểu đồ tròn, biểu đồ đường thẳng. Các loại biểu đồ mà bạn gặp thường rơi vào một trong bốn kiểu sau: Advanced, Group, Cross-Tab và OLAP, như hình 7.3.



Hình 7.3: Các dạng biểu đồ Crysral Report.

2.1. Biểu đồ Advanced (Biểu đồ cấp cao)

Biểu đồ Advanced làm việc giống như các biểu đồ mà bạn đã từng học ở trường: bạn vẽ một biểu đồ hoặc đồ thị dựa vào giá trị x và y.

2.2. Biểu đồ Group (Biểu đồ nhóm)

Biểu đồ nhóm thường được sử dụng trong các báo biểu, có thể được sử dụng bất cứ nơi nào mà bạn có thể chèn một nhóm vào trong báo biểu và tạo một Field tổng gộp dựa vào nhóm đó (Sum, Average, v.v...). Một

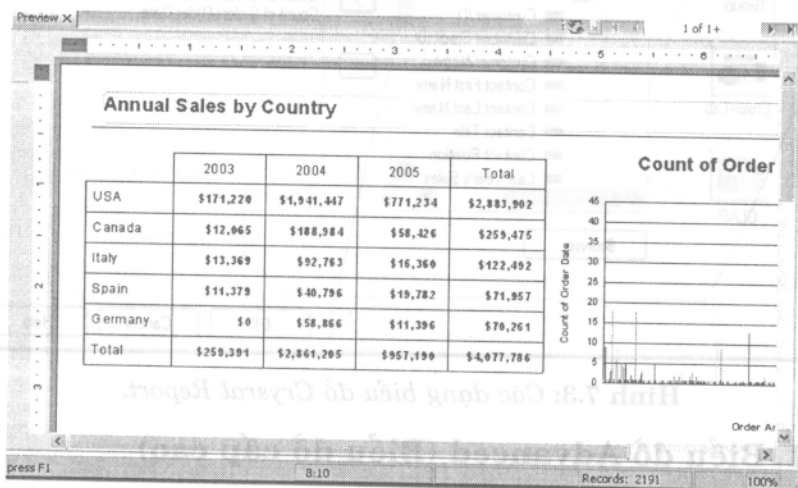
biểu đồ nhóm có thể xuất hiện một lần, đại diện dữ liệu trong toàn bộ báo biểu hoặc bạn có thể trình bày một biểu đồ cho mỗi nhóm.

Các biểu đồ nhóm có thể được sử dụng cho khả năng Drill-Through (duyet sâu thông tin), bạn có thể bắt đầu với một đồ thị ở mức cao nhất của dữ liệu (toàn bộ báo biểu) và xem dữ liệu phân cấp qua các nhóm khác. Đồ thị sẽ được trình bày theo mỗi nhóm, luồng thông tin sẽ dần hiện ra chi tiết khi bạn lần sâu xuống theo các trục của biểu đồ. Bạn có thể tham khảo ví dụ của báo biểu Drill-Through trong thư mục Project trên đĩa CD-ROM.

Nên nhớ rằng sự khác nhau giữa những biểu đồ Advanced và biểu đồ nhóm là biểu đồ Advanced không yêu cầu Group hoặc một Field tổng gộp có trong báo biểu. Bình thường, biểu đồ Advanced được sử dụng với các Field công thức.

2.3. Biểu đồ Cross-Tab

Dữ liệu trong báo biểu Cross-Tab có thể dùng để tạo biểu đồ theo hai trục tương ứng với giá trị của dòng và cột. Ví dụ hình 7.4 là một báo biểu Cross-Tab và biểu đồ bên cạnh.



Hình 7.4: Lưới Cross-tab và biểu đồ.

2.4. Biểu đồ khung lưới OLAP

Biểu đồ khung lưới OLAP tương tự như Cross-Tab, chỉ khác là OLAP trình bày dữ liệu đa hướng, trái ngược với Cross-Tab chỉ thể hiện dữ liệu quan hệ. Ngoài những sự khác biệt đó, chúng xử lý hoàn toàn giống nhau và chia sẻ tất cả các thuộc tính định dạng với nhau.

2.5. Thêm biểu đồ Advanced vào báo biểu

Biểu đồ Advanced được vẽ dựa vào giá trị x và y. Trong ví dụ sau chúng ta sẽ tạo một biểu đồ Advanced sử dụng báo biểu Order Listing từ CD ROM.

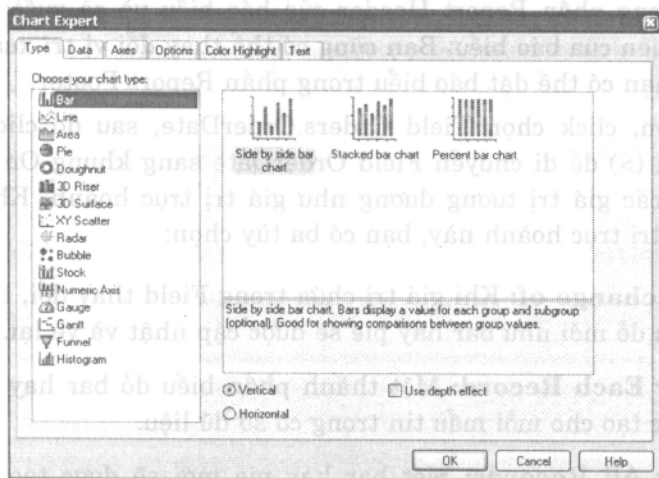
1. Mở Crystal Reports và mở báo biểu Orderlisting.rpt từ CD ROM, như hình 7.5.



Order Date	Customer Name	City	Region
12/02/2000	City Cyclists	Sterling Heights	MI
12/02/2000	Deals on Wheels	Dekalb	IL
12/02/2000	Warsaw Sports, Inc.	Warsaw	Warsaw
12/02/2000	Bikes and Trikes	Sterling Heights	MI
12/03/2000	SAB Mountain	Bern	Cantons
12/03/2000	Poser Cycles	Eden Prairie	MN
12/03/2000	Spokes	Dekalb	IL
12/03/2000	Clean Air Transportation Co.	Conshohocken	PA
12/03/2000	Extreme Cycling	Clearwater	FL
12/03/2000	Cyclopath	Kingston	RI
12/03/2000	BBS Pty	London	Greater London
12/03/2000	Piccolo	Salzburg	Salzkammergut
12/03/2000	Pedals Inc.	Madison	WI
12/04/2000	Spokes 'N' Wheels Ltd.	Des Moines	IA
12/04/2000	Cycle City Rome	Torino	Piedmonte
12/04/2000	SAB Mountain	Bern	Cantons
12/05/2000	Tyred Out	Santa Ana	CA
12/05/2000	Has Been Bikes (consignment)	Kingston	RI

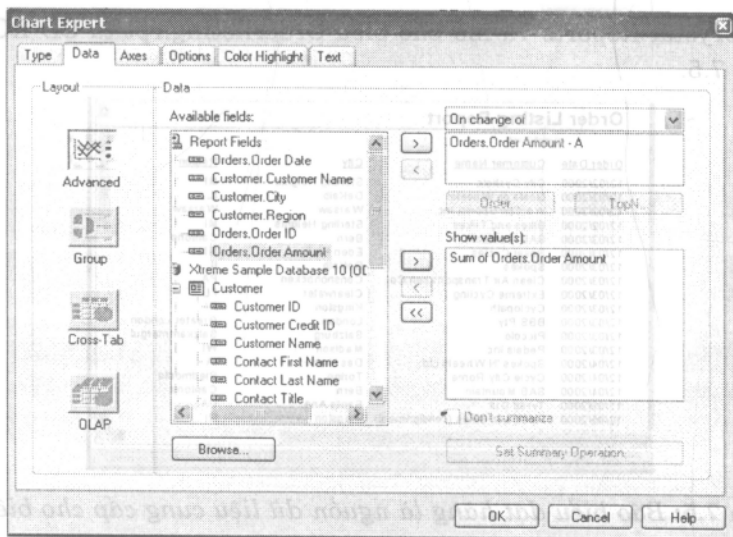
Hình 7.5: Báo biểu đặt hàng là nguồn dữ liệu cung cấp cho biểu đồ Advanced.

2. Chuyển báo biểu của bạn sang Tab Design, sau đó click Inset | Chart. Đặt biểu đồ vào mục Report Header và hộp thoại Chart Expert sẽ hiển thị như hình 7.6.



Hình 7.6: Hộp thoại chứa kiểu biểu đồ.

- Click vào Tab Type, chọn kiểu biểu đồ là Bar và sau đó click vào biểu tượng **Side by Side Chart** trong khung bên phải.
- Ở phía trên hộp thoại click vào tab Data. Hộp thoại sẽ có dạng như hình 7.7.

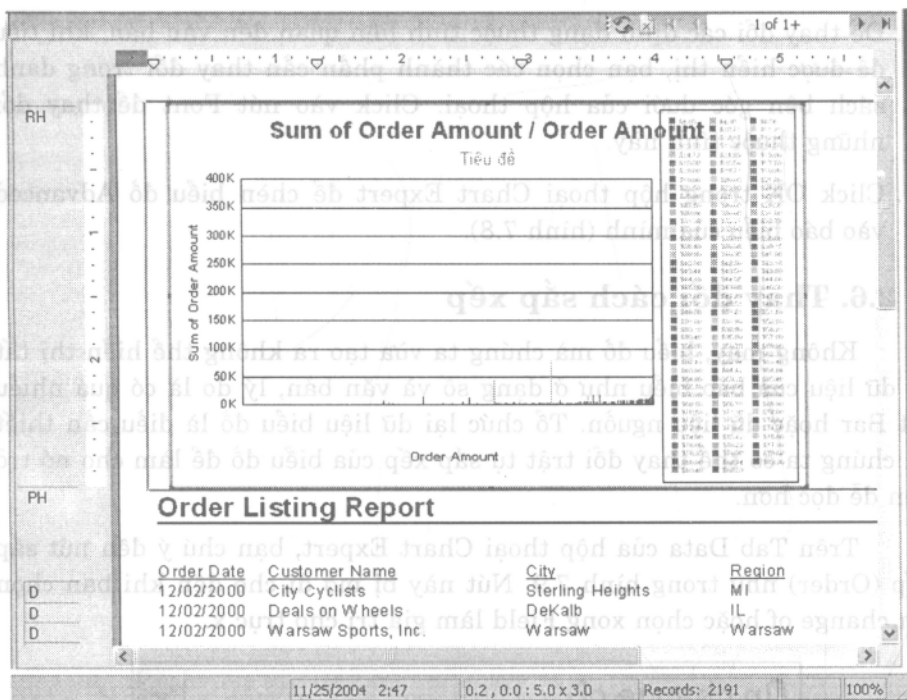


Hình 7.7: Thiết lập cho biểu đồ Advanced.

Trong ví dụ này, chúng ta sẽ tạo một biểu đồ Advanced hiển thị danh sách đơn đặt hàng được sắp xếp theo ngày đặt hàng. Biểu đồ sẽ hiển thị trong phần Report Header của báo biểu và sẽ xuất hiện trong trang đầu tiên của báo biểu. Bạn cũng có thể thay đổi vị trí xuất hiện của nó, chẳng hạn có thể đặt báo biểu trong phần Report Footer.

- Đầu tiên, click chọn Field Orders.OrderDate, sau đó click vào dấu mũi tên (>) để di chuyển Field OrderDate sang khung On change of, đây là các giá trị tương đương như giá trị trực hoành. Khi làm việc với giá trị trực hoành này, bạn có ba tùy chọn:

- On change of:** Khi giá trị chứa trong Field thay đổi, thành phần biểu đồ mới như bar hay pie sẽ được cập nhật và vẽ lại.
- For Each Record:** Một thành phần biểu đồ bar hay pie mới sẽ được tạo cho mỗi mẫu tin trong cơ sở dữ liệu.
- For All Records:** Một bar hay pie mới sẽ được tạo cho tất cả những mẫu tin trong cơ sở dữ liệu.



Hình 7.8: Báo biểu hoàn tất với biểu đồ advanced được thêm vào.

- Do biểu đồ trong ví dụ này là biểu đồ **Bar** để hiển thị Field Order Date, bạn sử dụng danh sách sổ xuống để chọn On change of.
- Tiếp theo, click chọn Field Orders.Order Amount, sau đó click vào mũi tên (<) phía dưới nhãn Show Values để di chuyển Field Orders.Order Amount vào danh sách, nó tương đương giá trị y trong biểu đồ của bạn

Ghi chú: Lưu ý rằng, bạn có thể chọn nhiều giá trị x trong đồ thị Advanced. Điều này cho phép bạn tạo các đồ thị phức tạp, đòi hỏi nhiều giá trị trên trục x, nhưng không phải đồ thị nào cũng hỗ trợ đa giá trị x.

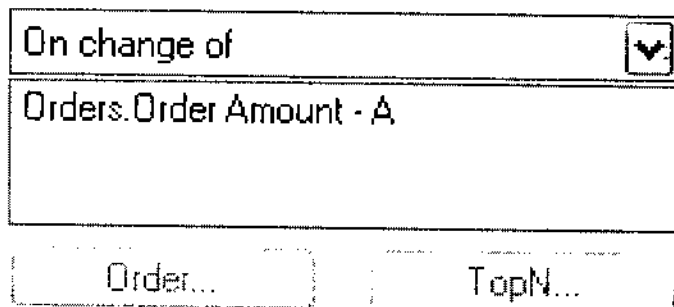
- Click vào tab Text để đặt tiêu đề cho biểu đồ của bạn. Như với các kiểu đồ thị khác, Crystal Reports tự tạo các tiêu đề cho biểu đồ, nhưng bạn có thể thay đổi và định dạng lại các tiêu đề cho biểu đồ theo ý mình.
- Để ghi đề lên các giá trị mặc định, bỏ chọn ô Auto Text và nhập vào tên tiêu đề cho biểu đồ.

10. Để thay đổi các định dạng thuộc tính liên quan đến văn bản, khi tiêu đề được hiển thị, bạn chọn các thành phần cần thay đổi trong danh sách bên góc dưới của hộp thoại. Click vào nút Font để thay đổi những thuộc tính này.
11. Click OK trong hộp thoại Chart Expert để chèn biểu đồ Advanced vào báo biểu của mình (hình 7.8).

2.6. Thay đổi cách sắp xếp

Không may, biểu đồ mà chúng ta vừa tạo ra không thể hiển thị tất cả dữ liệu của báo biểu như ở dạng số và văn bản, lý do là có quá nhiều cột Bar hoặc dữ liệu nguồn. Tổ chức lại dữ liệu biểu đồ là điều cần thiết và chúng ta có thể thay đổi trật tự sắp xếp của biểu đồ để làm cho nó trở nên dễ đọc hơn.

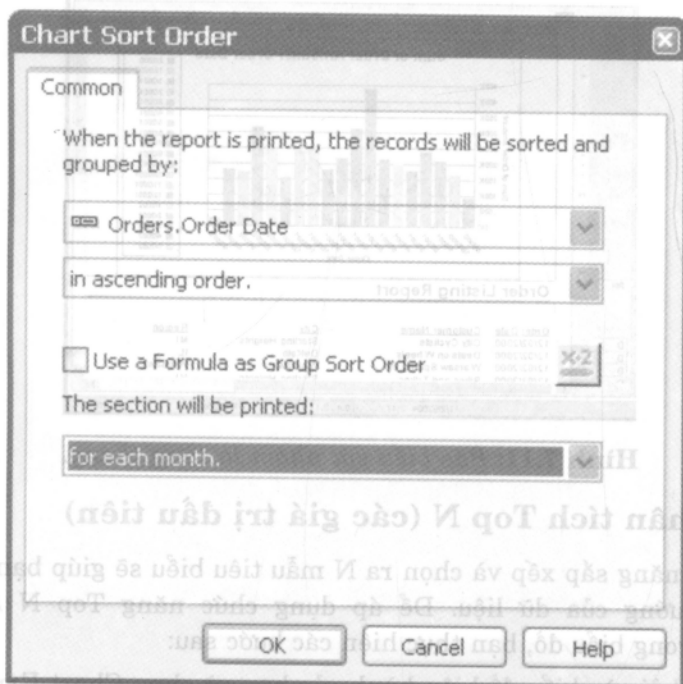
Trên Tab Data của hộp thoại Chart Expert, bạn chú ý đến nút sắp xếp (Order) như trong hình 7.9. Nút này bị mờ đi cho đến khi bạn chọn On change of hoặc chọn xong Field làm giá trị cho trục x.



Hình 7.9: Chọn dữ liệu cho biểu đồ.

Bạn có thể sử dụng nút này để sắp xếp và tổng kết dữ liệu xuất hiện trong biểu đồ, hãy thực hiện các bước như sau:

1. Từ báo biểu Order Listing chúng ta đã thực hiện ở trên, click phải vào báo biểu Advanced mà chúng ta vừa tạo và chọn Chart Expert từ menu tắt để mở hộp thoại Chart Expert.
2. Click vào tab Data.
3. Xác định Field Orders.Order.Date trong hộp bên dưới danh sách On change of và click chọn nó.
4. Click vào nút Order để mở hộp thoại Sort Order như hình 7.10.



Hình 7.10: Tùy chọn sắp xếp biểu đồ.

Có bốn tùy chọn sắp xếp:

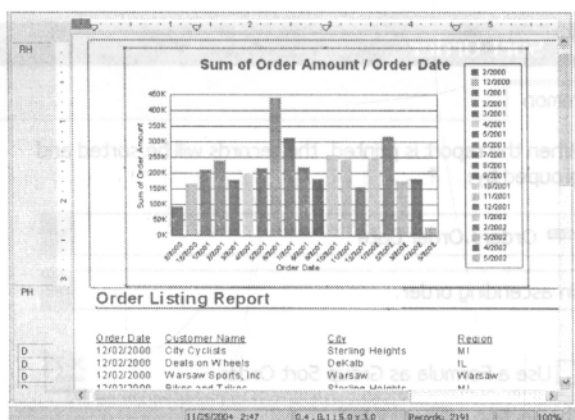
- **Ascending:** Sắp xếp tăng từ A to Z, 0 to 9.
- **Descending:** Sắp xếp giảm Z to A, 9 to 0.
- **Specified:** Sắp xếp theo thứ tự riêng.
- **Original:** Giữ nguyên thứ tự mặc định của dữ liệu.

5. Mặc định của tùy chọn sắp xếp là tăng dần.

Do ta vừa tạo biểu đồ Advanced trên Field ngày tháng (Order Date), một ô danh sách thứ ba xuất hiện với các tùy chọn nhóm theo khoảng thời gian như each day, week, month, v.v...

6. Để biểu đồ dễ đọc hơn, bạn chọn khoảng thời gian là For each Month và click OK.

7. Click OK lần nữa từ hộp thoại Chart Expert để trở về Report Designer. Bây giờ báo biểu của bạn sẽ trông giống như hình 7.11.



Hình 7.11: Báo biểu với nhóm là từng tháng.

2.7. Phân tích Top N (các giá trị đầu tiên)

Khả năng sắp xếp và chọn ra N mẫu tiêu biểu sẽ giúp bạn xác định được xu hướng của dữ liệu. Để áp dụng chức năng Top N / Sort all Analysis trong biểu đồ, bạn thực hiện các bước sau:

1. Click phải vào biểu đồ hiện hành của bạn và chọn Chart Expert.
 2. Trong hộp thoại Chart Expert, click tab Data.
 3. Tìm Field dùng sắp xếp để chọn ra N mẫu tin trong danh sách On Change Of, click chọn nó.
 4. Khi Field đã được chọn, nút Top N sẽ có hiệu lực. Click vào nút này để mở hộp thoại Top N / Sort.
- **Top N:** Nhập vào giá trị N để xác định số giá trị đầu tiên cần lấy dựa trên Field giá trị y mà bạn chọn.
 - **Bottom N:** Nhập vào giá trị N để xác định số giá trị cuối cùng cần lấy dựa trên Field giá trị y mà bạn chọn.
 - **Sort All:** Chọn tùy chọn này để sắp xếp tất cả dữ liệu hoặc là tăng hoặc là giảm dựa vào giá trị y.

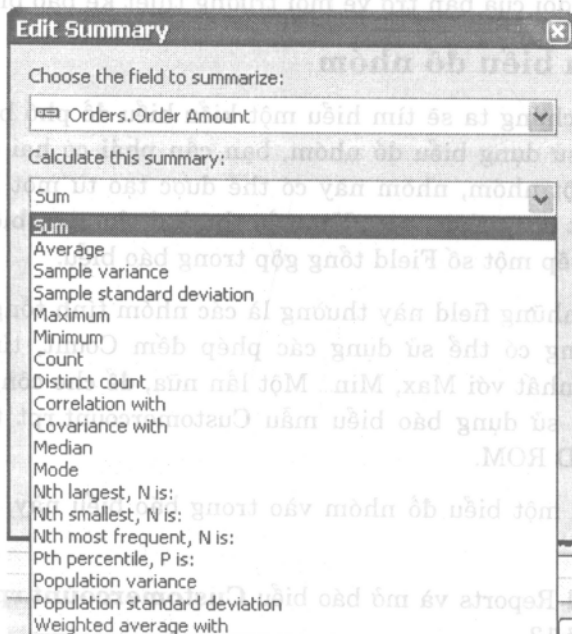
Với Top N và Bottom N, bạn cũng có thể xác định cách các giá trị N thể hiện. Khi phân tích báo biểu với Top N / Bottom N, bạn đã loại bỏ những giá trị không thỏa mãn bằng cách giữ chúng trong một nhóm khác.

5. Khi đã hoàn thành việc thiết lập tùy chọn Top N / Sorting, click OK để chấp nhận thay đổi của bạn và trở về môi trường thiết kế. Đồ thị của bạn bây giờ sẽ được cập nhật theo những tùy chọn mà bạn đã chỉ định.

2.8. Thay đổi toán tử tổng gộp

Một đặc tính khác của báo biểu mà chúng ta có thể sử dụng trong biểu đồ là toán tử tổng gộp đã học qua trong chương 4. Mặc định, một biểu đồ kiểu Advanced sẽ sử dụng một toán tử tổng gộp Sum, nhưng bạn có thể thay đổi toán tử này bằng những toán tử tổng gộp mà Crystal Reports hỗ trợ. Để thay đổi toán tử tổng gộp, bạn thực hiện các bước sau:

1. Click phải vào biểu đồ hiện hành của bạn và chọn Chart Expert.
2. Click vào Field tổng gộp mà bạn muốn thay đổi phép tính.
3. Kích vào nút Set Summary Operation, hộp thoại như hình 7.12 sẽ hiển thị. Bạn có thể chọn các toán tử tính tổng gộp từ danh sách.



Hình 7.12: Toán tử Summary có thể sử dụng trong biểu đồ.

- **Sum:** Tính tổng, sử dụng cho kiểu dữ liệu Numeric, Currency.
- **Average:** Tính trung bình cộng, sử dụng cho kiểu dữ liệu Numeric, Currency.

- **Minimum:** Xác định giá trị nhỏ nhất của Field trong cơ sở dữ liệu, sử dụng cho các kiểu dữ liệu Numeric, Currency, String và Date.
 - **Maximum:** Xác định giá trị lớn nhất của Field trong cơ sở dữ liệu, sử dụng cho các kiểu dữ liệu Numeric, Currency, String, Date
 - **Count:** Đếm những giá trị của Field trong cơ sở dữ liệu, sử dụng cho tất cả các kiểu dữ liệu.
 - **Distinct Count:** Tương tự như Count, trừ đi các giá trị giống nhau (chỉ đếm một lần).
4. Click OK để chấp nhận thay đổi kiểu tổng gộp của bạn và trở về hộp thoại Chart Expert.
 5. Khi đã hoàn thành việc chỉnh sửa biểu đồ của, click OK để chấp nhận thay đổi của bạn trở về môi trường thiết kế báo biểu.

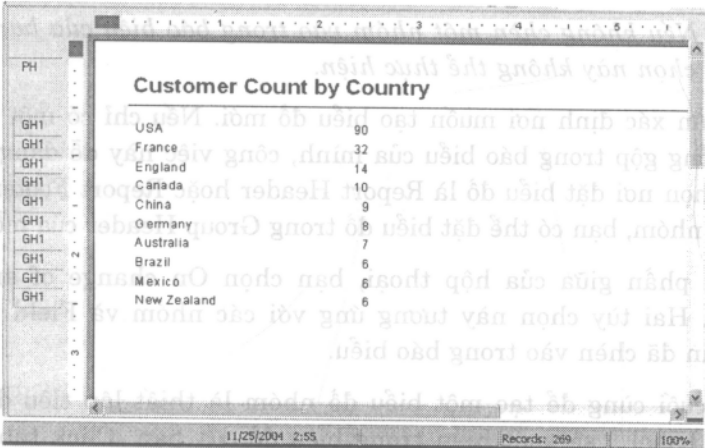
2.9. Chèn biểu đồ nhóm

Bây giờ chúng ta sẽ tìm hiểu một kiểu biểu đồ phổ biến khác: biểu đồ nhóm. Để sử dụng biểu đồ nhóm, bạn cần phải có hai điều kiện sau: Đầu tiên là một nhóm, nhóm này có thể được tạo từ một Field trong cơ sở dữ liệu, một công thức, v.v... Yêu cầu thứ hai cho một biểu đồ nhóm là bạn phải sắp xếp một số Field tổng gộp trong báo biểu.

Mặc dù những field này thường là các nhóm tính tổng trên field số, nhưng bạn cũng có thể sử dụng các phép đếm Count, tính giá trị lớn nhất hay nhỏ nhất với Max, Min... Một lần nữa, để cho công việc dễ dàng hơn, chúng ta sử dụng báo biểu mẫu Customercount.rpt trong thư mục Project trên CD ROM.

Để chèn một biểu đồ nhóm vào trong báo biểu này, bạn thực hiện các bước sau:

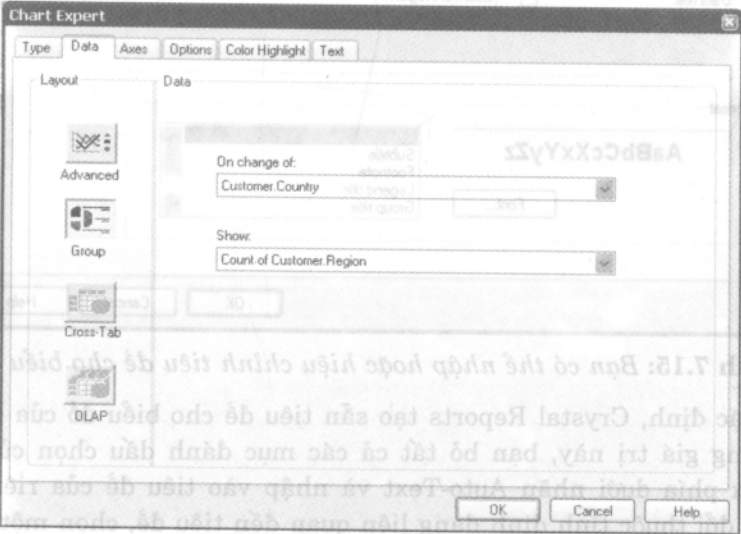
1. Mở Crystal Reports và mở báo biểu **Customercount.rpt** từ CD ROM, như hình 7.13.
2. Chuyển báo biểu của bạn sang Tab Design, click Insert | Chart và click vào mục Report Header trong biểu đồ của bạn. Biểu đồ sẽ tự động xuất hiện với các thiết đặt mặc định của nó. Để thay đổi những thiết đặt này, click phải vào biểu đồ và chọn Chart Expert.



Hình 7.13: Báo biểu sử dụng với biểu đồ nhóm.

Bước đầu tiên trong việc tùy biến một biểu đồ nhóm là lựa chọn một kiểu biểu đồ mà bạn muốn tạo.

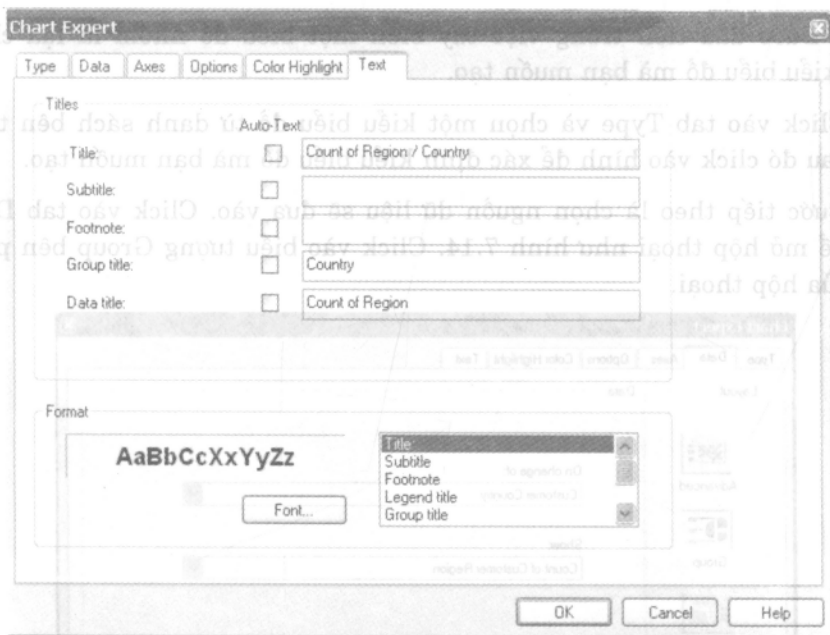
- 3. Click vào tab Type và chọn một kiểu biểu đồ từ danh sách bên trái, sau đó click vào hình để xác định kiểu biểu đồ mà bạn muốn tạo.
- 4. Bước tiếp theo là chọn nguồn dữ liệu sẽ đưa vào. Click vào tab Data để mở hộp thoại như hình 7.14. Click vào biểu tượng Group bên phải của hộp thoại.



Hình 7.14: Bạn có thể chọn nguồn dữ liệu cho biểu đồ nhóm.

Ghi chú: Nếu không chèn một nhóm vào trong báo biểu của bạn, thì tùy chọn này không thể thực hiện.

- Bạn cần xác định nơi muốn tạo biểu đồ mới. Nếu chỉ có một nhóm và một tổng gộp trong báo biểu của mình, công việc này dễ dàng, bạn chỉ việc chọn nơi đặt biểu đồ là Report Header hoặc Report Footer. Nếu có nhiều nhóm, bạn có thể đặt biểu đồ trong Group Header của mỗi nhóm.
- Trong phần giữa của hộp thoại, bạn chọn On change of and Show Value. Hai tùy chọn này tương ứng với các nhóm và Field tổng gộp mà bạn đã chèn vào trong báo biểu.
- Bước cuối cùng để tạo một biểu đồ nhóm là thiết lập tiêu đề (tức là Title, Label...) sẽ xuất hiện trong biểu đồ của bạn. Click tab Text để hiển thị hộp thoại như hình 7.15.



Hình 7.15: Bạn có thể nhập hoặc hiệu chỉnh tiêu đề cho biểu đồ.

Mặc định, Crystal Reports tạo sẵn tiêu đề cho biểu đồ của bạn. Để sửa những giá trị này, bạn bỏ tất cả các mục đánh dấu chọn của các ô checkbox phía dưới nhãn Auto-Text và nhập vào tiêu đề của riêng bạn. Để thay đổi thuộc tính định dạng liên quan đến tiêu đề, chọn một mục từ danh sách phía cuối góc phải hộp thoại và click vào nút Font để thay đổi thuộc tính.

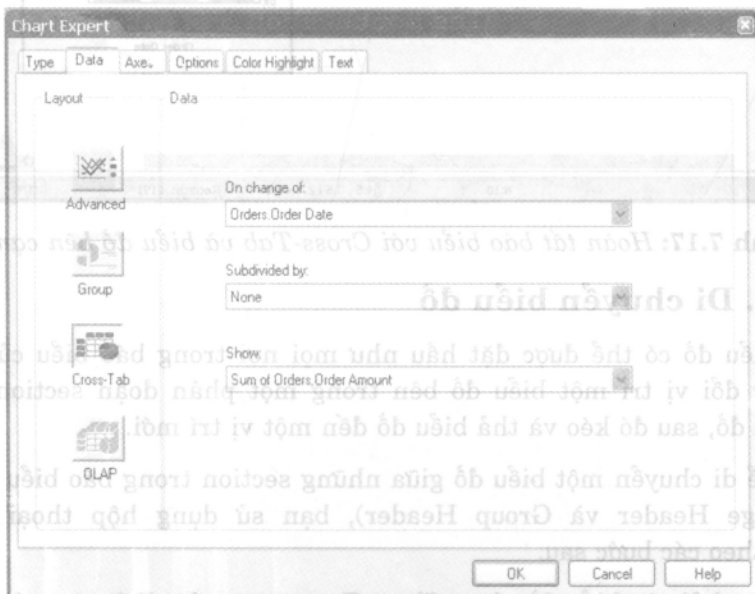
8. Bây giờ click OK, các tùy chọn biểu đồ mới của bạn được chấp nhận và thêm vào trong báo biểu.

2.10. Chèn thêm biểu đồ Cross-Tab

Bạn có thể chèn một biểu đồ dựa trên dữ liệu kết xuất của bảng báo biểu Cross-Tab. Trong ví dụ sau, chúng ta sẽ xây dựng biểu đồ cho báo biểu Crosstab.rpt nằm trong thư mục Project của đĩa CD ROM. Để thêm một biểu đồ Cross-Tab vào báo biểu, bạn thực hiện các bước sau:

1. Mở Crystal Reports và mở báo biểu Crosstab.rpt từ CD ROM.
 - a) Xem trước nội dung báo biểu để chắc chắn rằng dữ liệu kết xuất đầy đủ ở trên các dòng, cột và có cả cột tổng cộng.
 - b) Chuyển sang tab Design của báo biểu, click Insert | Chart, nhấp vào nơi chứa kiểu biểu đồ của hộp thoại Chart Expert.

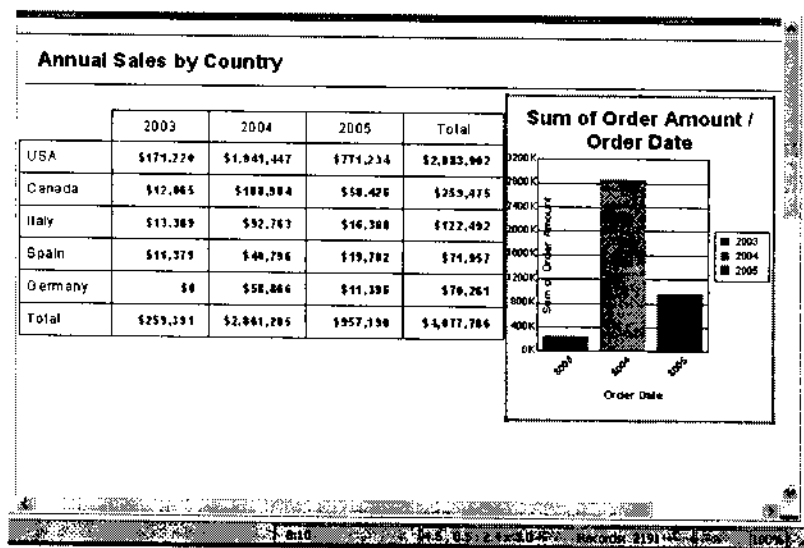
Ghi chú: Crystal Reports tự động thiết lập các thuộc tính biểu đồ mặc định để giảm bớt một số yêu cầu thiết lập biểu đồ. Nếu không muốn sử dụng các giá trị thiết lập mặc định này, bạn có thể bỏ nút check Automatically Set Chart Options trong hộp thoại Chart Expert. Những tab bổ sung mới sẽ xuất hiện trong hộp thoại Chart Expert.



Hình 7.16: Tùy chọn biểu đồ Cross-Tab.

- Click vào tab Data để tiến hành các bước tiếp theo của hộp thoại Chart Expert. Trên hộp thoại Chart Expert, Click vào kiểu Cross-Tab và chọn cách sắp xếp biểu đồ của bạn. Nếu báo biểu của bạn chỉ có một Cross-Tab, bạn có thể đặt biểu đồ trong Report Header hoặc Report Footer.
- Click vào Tab Text để tiến đến bước cuối cùng trong hộp thoại Chart Expert, và chọn tiêu đề cho biểu đồ cùng các định dạng cho đối tượng.
- Khi đã hoàn thành việc sửa tiêu đề của biểu đồ, click OK quay về báo biểu.

Báo biểu của bạn bây giờ sẽ có hình dạng như hình 7.17.



Hình 7.17: Hoàn tất báo biểu với Cross-Tab và biểu đồ bên cạnh.

2.11. Di chuyển biểu đồ

Biểu đồ có thể được đặt hầu như mọi nơi trong báo biểu của bạn. Để thay đổi vị trí một biểu đồ bên trong một phân đoạn section, click vào biểu đồ, sau đó kéo và thả biểu đồ đến một vị trí mới.

Để di chuyển một biểu đồ giữa những section trong báo biểu (ví dụ, giữa Page Header và Group Header), bạn sử dụng hộp thoại Chart Expert theo các bước sau:

- Click phải vào biểu đồ, chọn Chart Expert, sau đó click vào tab Data.
- Sử dụng ô danh sách phía trên hộp thoại, chọn section mới cho biểu đồ.

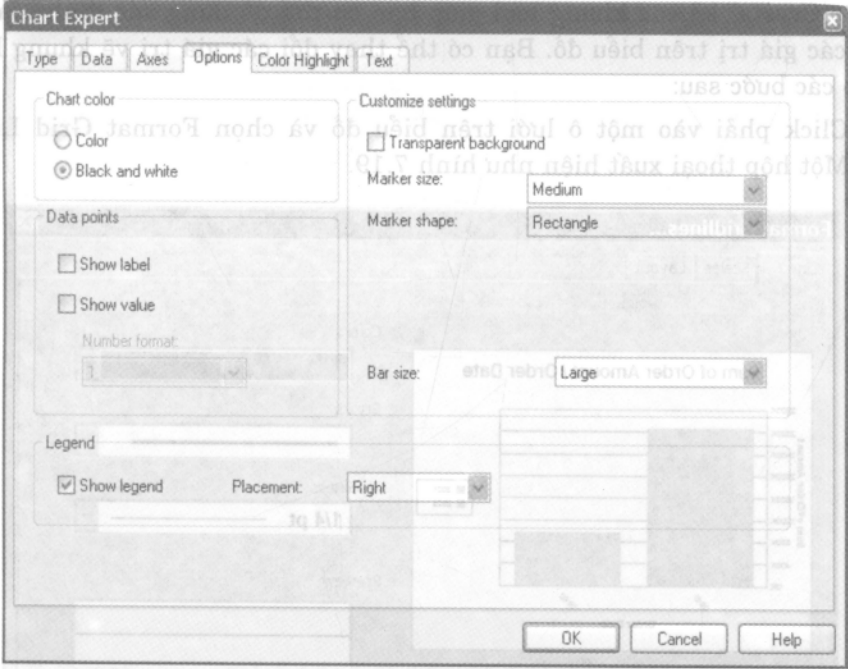
3. Click OK để chấp nhận sự thay đổi. Biểu đồ của bạn bây giờ sẽ xuất hiện trong section mới mà bạn đã chọn.

2.12. Thay đổi biểu đồ màu thành biểu đồ trắng và đen

Mặc định, biểu đồ trong Crystal Reports được vẽ với các màu sắc khác nhau. Tuy nhiên, bạn có thể chọn chỉ sử dụng màu nền trắng và đen để kết quả ra máy in rõ ràng hơn.

Để chuyển một biểu đồ sang màu trắng đen, bạn thực hiện các bước sau:

- 1. Click phải trên biểu đồ mà bạn muốn đổi màu và click vào Format | Chart Expert từ menu tắt.
- 2. Click vào tab Option để mở hộp thoại như hình 7.18.



Hình 7.18: Một số tùy chọn biểu đồ có sẵn để bạn có thể thao tác trong Expert.

3. Click vào nút radio Black and White, sau đó click OK để chấp nhận thay đổi của bạn. Biểu đồ của bạn bây giờ sẽ thay đổi chỉ có màu đen và xám.

2.13. Hiện thị chú giải

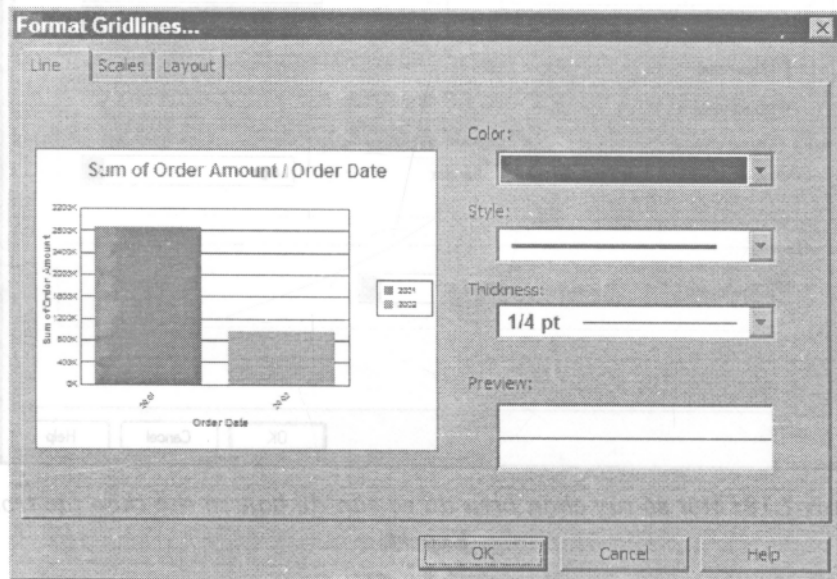
Crystal Reports tạo ra chú giải cho biểu đồ. Để thao tác với các chú giải trong Chart Expert, bạn thực hiện theo các bước sau:

1. Click phải trên biểu đồ mà bạn muốn làm việc và click vào Format | Chart Expert từ menu tắt.
2. Click vào Tab Type trên hộp thoại Chart Expert, bỏ chọn mục Automatically Set Chart Options và click vào Tab Option.
3. Định vị mục Lenged và chọn checkbox Show Lenged để đặt lời chú giải.
4. Sử dụng danh sách sổ xuống kế bên để chọn nơi đặt chú giải (Right, left hoặc bottom).
5. Click OK để chấp nhận thay đổi của bạn.

2.14. Điều khiển ô lưới và tỷ lệ đo

Crystal kẻ các khung lưới giúp bạn có thể dễ dàng xác định tọa độ của các giá trị trên biểu đồ. Bạn có thể thay đổi các giá trị về khung lưới theo các bước sau:

1. Click phải vào một ô lưới trên biểu đồ và chọn Format Grid Line. Một hộp thoại xuất hiện như hình 7.19.



Hình 7.19: Bạn có thể thiết lập định dạng lưới bằng cách sử dụng các tùy chọn được hỗ trợ.

2. Click chọn Tab Scale hoặc Layout.
3. Trên Tab Layout, chọn Show Gridlines để hiển thị khung lưới.
4. Click vào tab Scale để đặt tùy chọn cho tỷ lệ chia trên trục.
5. Khi đã hoàn thành việc thiết lập tùy chọn lưới và tỷ lệ, click OK để chấp nhận thay đổi của bạn.

2.15. Thay đổi kiểu biểu đồ

Bạn có thể áp dụng một số kiểu biểu đồ khác nhau cho báo biểu của mình. Ngoài những kiểu biểu đồ chuẩn như Bar, Pie bạn còn có nhiều kiểu biểu đồ chuyên dùng cho thống kê như biểu đồ rada và biểu đồ nổi bọt (bubble graphic), biểu đồ mảng ba chiều (three-dimensional) và nhiều loại biểu đồ khác. Để thay đổi kiểu biểu đồ, bạn thực hiện theo các bước sau:

1. Click phải vào biểu đồ bạn muốn thay đổi, chọn Load Template từ menu tắt.
2. Từ Tab Gallery, chọn một kiểu biểu đồ mới bằng cách click vào biểu tượng các biểu đồ trong danh sách có sẵn.
3. Click OK để chấp nhận thay đổi của bạn và trở về báo biểu.

Ngoài những tùy chọn cơ bản, Crystal Reports còn có một số định dạng nâng cao dành cho biểu đồ và dữ liệu. Các thuộc tính định dạng này bạn có thể sử dụng bằng cách click phải vào một biểu đồ và chọn Chart Options.

2.16. Chỉnh kích thước Font và màu cho biểu đồ

Một trong các tác vụ định dạng thông dụng nhất bạn sẽ dùng đến là thay đổi Font, kích thước và màu sắc cho các thành phần biểu đồ. Để thực hiện những thay đổi này, bạn làm theo các bước sau:

1. Tìm thành phần biểu đồ mà bạn muốn thay đổi, click phải vào biểu đồ và chọn Format XXX để mở hộp thoại Formatting.
2. Sử dụng hộp thoại như hình 7.20, chọn Font, kích thước, kiểu và màu cho đối tượng.
3. Click OK để trở về báo biểu.



3. TÓM TẮT

Chương này chúng ta đã học về kỹ thuật xây dựng biểu đồ trình diễn số liệu báo biểu dưới dạng hình ảnh. Bất chấp dù bạn có 10 hoặc 10.000 mẫu tin, biểu đồ vẫn là cách đơn giản nhất để tổng hợp và hiển thị thông tin trong báo biểu khiến người dùng dễ nhận biết và phân tích nhất. Trong chương sau, chúng ta sẽ bắt đầu tìm hiểu về những kỹ thuật tích hợp báo biểu với các ứng dụng Windows và Web.

Chương 8:

TÍCH HỢP BÁO BIỂU

- ❖ *Tổng quan.*
- ❖ *Chọn phương pháp tích hợp.*
- ❖ *Chọn nguồn dữ liệu.*
- ❖ *Chọn cấu trúc báo biểu.*
- ❖ *Tóm tắt.*

1. TỔNG QUAN

Bạn đã biết cách thiết kế các loại báo biểu, phần tiếp theo của các chương trong giáo trình này sẽ hướng dẫn bạn cách tích hợp báo biểu vào những loại ứng dụng khác nhau. Báo biểu của bạn có thể tích hợp với ứng dụng Windows, Web và Java. Chương này chủ yếu giới thiệu với bạn các phương pháp tích hợp, lựa chọn nguồn dữ liệu cho báo biểu, những kiến trúc tích hợp nào được xem là tốt nhất.

2. CHỌN PHƯƠNG PHÁP TÍCH HỢP

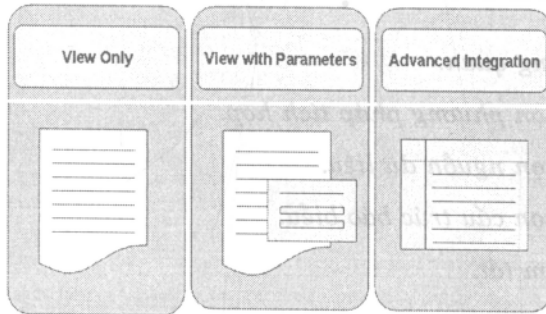
Thường thì các nhà phát triển ứng dụng tích hợp các báo biểu vào trong chương trình dựa vào nguồn dữ liệu là cơ sở dữ liệu mà chương trình sử dụng, chương trình cung cấp cho người dùng các mục chọn menu để chọn loại báo biểu cần xem. Hình 8.1 là mô hình chung nhất khi phát triển ứng dụng có sử dụng báo biểu. Thông thường người dùng sẽ chọn tên báo biểu cần xem từ menu, trình Viewer sẽ hiển thị nội dung báo biểu (View Only). Với một số báo biểu, người dùng có thể nhập vào tham số thông qua hộp thoại (View with Parameters). Với những ứng dụng cao cấp hơn, việc tích hợp còn cho phép người dùng tương tác, lọc, drill-down thông tin trên báo biểu.

2.1. Report Viewer

Viewer là một công cụ để hiển thị nội dung báo biểu. Ví dụ, để xem báo biểu từ một ứng dụng Windows được tạo với công cụ phát triển Visual Studio 2005, trước tiên bạn kéo điều khiển ReportViewer vào cửa

sổ Form và nhập dòng mã sau để gán báo biểu cho Viewer hiển thị nội dung báo biểu sales.rpt:

`CrystalReportViewer1.ReportSource="c:\sales.rpt"`



Hình 8.1: Các tùy chọn tích hợp báo biểu phổ biến.

Người sử dụng có thể thực thi và xem trước (Preview) nội dung báo biểu trên Viewer như hình 8.2.



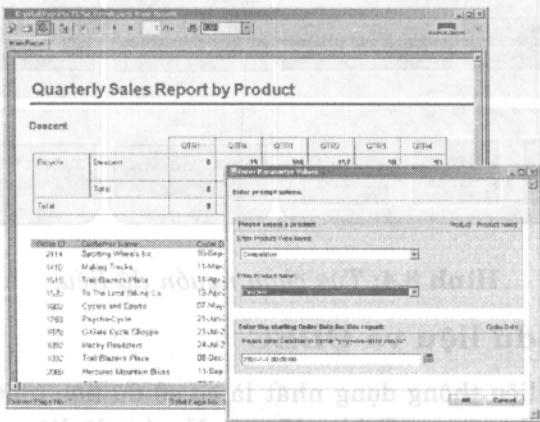
Hình 8.2: Report Viewer chuẩn cho các ứng dụng Windows.

Mỗi môi trường phát triển ứng dụng khác nhau sẽ có thành phần Viewer với một tập lệnh phong phú theo mô hình đối tượng, cho phép bạn điều khiển diện mạo và cách hiển thị nội dung báo biểu. Một Viewer thường bao gồm thanh công cụ với các nút lệnh cho phép nhảy trang, in, lọc, lưu file...

2.2. Sử dụng Report Viewer với tham số

Kiểu tích hợp kế tiếp được sử dụng khá rộng rãi là truyền tham số cho báo biểu. Trong trường hợp này, người phát triển ứng dụng sẽ tạo ra các báo biểu từ cơ sở dữ liệu với tham số được dùng để lọc nội dung báo biểu theo yêu cầu của người dùng cuối.

Hình 8.3 là một ví dụ của ứng dụng Windows hiển thị báo biểu trong Viewer với tham số cho phép người dùng nhập từ bên ngoài vào để làm điều kiện lọc cho báo biểu.



Hình 8.3: Báo biểu với tham số trong ứng dụng Windows.

2.3. Tích hợp nâng cao

Khi người phát triển ứng dụng mong muốn điều khiển báo biểu ở mức sâu hơn, họ có thể bỏ qua Viewer hoặc chỉ dùng một phần chức năng của Viewer. Họ thường sử dụng các hàm lập trình API mà Crystal cung cấp để tạo báo biểu động, thực hiện những công việc đặc thù như nhúng Viewer vào một phần ứng dụng nào đó.

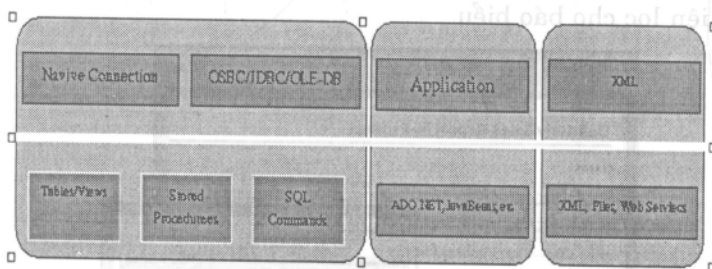
Trong cả ba phương pháp tích hợp trên thì phương pháp xem báo biểu với tham số chiếm đến 70%, phương pháp chỉ đơn thuần xem nội dung báo biểu mà không cần tham số chiếm 20% và phần còn lại là phương pháp tích hợp nâng cao không dùng đến Viewer (chiếm 10%).

Tại sao xem một báo biểu với tham số lại sử dụng nhiều như vậy? Lý do là dùng tham số bạn có thể linh động tạo ra nhiều báo biểu khác nhau. Nội dung báo biểu có thể thay đổi theo yêu cầu của người dùng cuối. Phương pháp không dùng Viewer ít được sử dụng vì không phải nhà

phát triển ứng dụng nào cũng rành về các hàm hệ thống API của Crystal.

3. LỰA CHỌN NGUỒN DỮ LIỆU

Sau khi đã chọn được phương pháp tích hợp báo biểu cho ứng dụng, vấn đề tiếp theo bạn cần quan tâm là các nguồn dữ liệu mà báo biểu sẽ sử dụng. Hình 8.4 là những nguồn dữ liệu mà Crystal Reports có thể hỗ trợ.



Hình 8.4: Tùy chọn nguồn dữ liệu.

3.1. Cơ sở dữ liệu và ODBC

Nguồn dữ liệu thông dụng nhất là cơ sở dữ liệu (database). Dữ liệu database có thể lưu trong Table, Views. Nguồn dữ liệu này có thể truy cập thông qua các trình điều khiển (driver) như ODBC, JDBC, OLE-DB hoặc qua trình điều khiển truy xuất trực tiếp (native) mà cơ sở dữ liệu cung cấp.

Crystal Reports cung cấp driver ODBC cũng như trình truy xuất trực tiếp native cho nhiều loại cơ sở dữ liệu thông dụng. Bảng dưới đây là danh sách các cơ sở dữ liệu thường sử dụng nhất và những phương pháp mà bạn có thể truy xuất thông qua driver của Crystal.

Database	Native	ODBC	JDBC	OLE-DB
Oracle 8.17	X	X	X	X
Oracle 9.2	X	X	X	X
Oracle 10.1	X	X	X	X
IBM DB2 8.1	X	X	X	X
IBM DB2 8.2	X	X	X	X
Sybase ASE 12.5	X	X		

<i>Teradata v2R5, R6</i>		X		
<i>Microsoft SQL Server 2003 SP3</i>		X	X	X
<i>Microsoft SQL Server 7.0 SP4</i>		X		X
<i>MySQL 4.0</i>		X	X	
<i>Microsoft Access 2000</i>	X	X		X
<i>Microsoft Access 2003</i>	X	X		X
<i>Microsoft Excel 2000</i>	X	X		X
<i>Microsoft Excel 2004</i>	X	X		X
<i>Microsoft Exchange</i>	X			
<i>ADO.Net</i>	X			
<i>XML</i>		X		
<i>All Other ODBC Drivers</i>		X		
<i>All Other JDBC Drivers</i>			X	

3.2. Dữ liệu ứng dụng

Nguồn dữ liệu phổ biến khác cho các báo biểu là dữ liệu bên trong các ứng dụng. Thường ứng dụng sẽ sử dụng những tập dữ liệu phức tạp có cấu trúc đặc thù riêng của mình. Chẳng hạn bạn có thể xây dựng một cấu trúc file để lưu địa chỉ khách hàng của riêng mình và nhờ Crystal đọc dữ liệu và in nội dung danh sách địa chỉ ra báo biểu. Không phải nguồn dữ liệu của ứng dụng nào cũng mở và có thể truy xuất được.

3.3. XML và những dịch vụ Web XML

Một nguồn dữ liệu nữa có thể dùng cho báo biểu là file XML và các dịch vụ Web XML. Crystal Reports XI bao gồm một bộ điều khiển XML linh hoạt có thể sử dụng để tạo báo biểu từ file XML cục bộ và trên máy chủ Web Server hay thông qua các dịch vụ XML Web. Phương pháp này mở ra một khả năng để làm việc với dữ liệu cả bên trong và bên ngoài tổ chức cũng như tạo khả năng giao tiếp cho các chương trình chưa được Crystal hỗ trợ truy xuất nguồn dữ liệu bên trong ứng dụng.

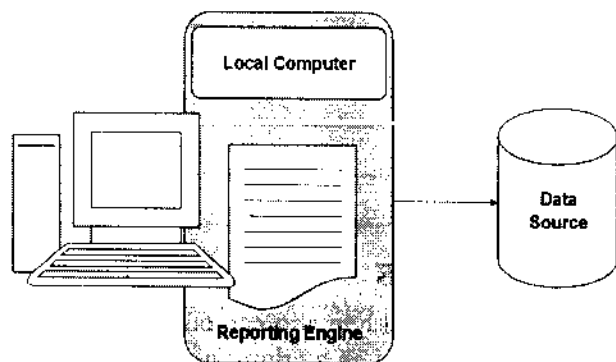
Nếu phải chọn giữa một trong số những phương pháp truy cập dữ liệu, tốt nhất nên chọn ODBC hoặc JDBC, OLE-DB, vì đây là phương pháp dễ sử dụng và phổ biến nhất.

4. CHỌN KIẾN TRÚC BÁO BIỂU

Mục đích chung nhất của hầu hết những nhà phát triển là tạo ra các ứng dụng có thể thay đổi dễ dàng (scalable) và truy cập bởi nhiều người sử dụng. Để làm được điều này bạn phải xây dựng kiến trúc của ứng dụng sao cho phù hợp. Dưới đây là những kiến trúc ứng dụng thường sử dụng các báo biểu của Crystal nhất.

4.1. Ứng dụng Thick-Client

Ứng dụng Thick-Client là những ứng dụng truyền thống chạy cục bộ trên máy PC của người sử dụng như hình 8.5.



Hình 8.5: Một ứng dụng Thick-Client.

Đa số người phát triển và người sử dụng đều quen thuộc với kiểu ứng dụng này do hầu hết các ứng dụng của Windows là Thick-Client. Mặc dù phần lớn những ứng dụng này chạy độc lập trên máy đơn nhưng ứng dụng Thick-Client đôi khi cũng được xem là loại ứng dụng khách-chủ (Client-Server) nếu chúng lưu cơ sở dữ liệu trên máy chủ thay vì trên máy cục bộ.

Ngay các báo biểu của Crystal Reports đa phần được thiết kế chạy cục bộ phục vụ cho ứng dụng Thick-Client, một số ứng dụng thương mại sử dụng kiến trúc này cũng đơn giản và rất dễ tích hợp.

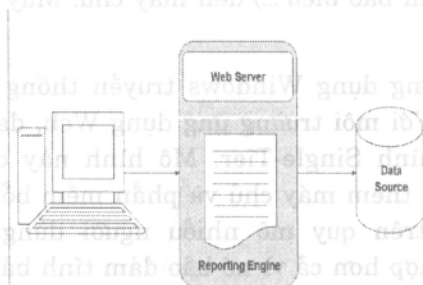
Một trong những thuận lợi khi phân phối báo biểu theo kiến trúc này là mọi yêu cầu xem báo biểu đều được thiết lập cục bộ. Thuận lợi khác là người sử dụng có thể xem và tương tác với báo biểu trên máy cục bộ không yêu cầu mạng hay kết nối với nguồn dữ liệu ở máy chủ.

Sự bất lợi của kiểu kiến trúc này cũng là do các báo biểu thực thi cục bộ cho nên nếu có thay đổi thì người sử dụng sẽ phải chờ các nhà phát triển gửi đến phiên bản cập nhật của báo biểu. Ngoài ra, việc phân phối một ứng dụng Thick-Client yêu cầu phân phối kèm theo một số thành phần của Crystal Reports, thường thì cài đặt những phiên bản Crystal khác nhau có thể dẫn đến những lỗi không tương thích về phiên bản.

Các ứng dụng Thick-Client tuy có những giới hạn nhưng thế mạnh là nó cho phép tạo ra các báo biểu những trực tiếp vào các ứng dụng. Hãy chọn một kiểu kiến trúc này nếu ứng dụng của bạn được thiết kế chủ yếu để chạy trên máy cục bộ đơn giản hóa các tác vụ đơn giản của người sử dụng.

4.2. Ứng dụng Single-Tier

Ứng dụng Single-Tier có thêm tầng xử lý phía máy chủ (thường là Web Server). Tất cả công việc xử lý được đặt tại trung tâm máy chủ (xem hình 8.6).

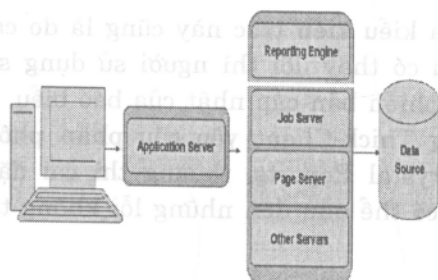


Hình 8.6: Ứng dụng Single-Tier.

Kiểu ứng dụng báo biểu này dần dần phát triển thành một tập hợp những công cụ mà Crystal Reports dùng hỗ trợ cho các ứng dụng Web. Nhưng một khuyết điểm trong mô hình Single-Tier truyền thống là tất cả công việc xử lý dồn trên máy chủ Web Server. Vì lý do này, các ứng dụng báo biểu có thể gây quá tải cho trình chủ Web Server do Web Server phải lo xử lý những tác vụ truy xuất tài nguyên khác. Để khắc phục, Crystal đưa ra mô hình ứng dụng báo biểu đa tầng.

4.3. Ứng dụng Multitiered (đa tầng)

Hình 8.7 là mô hình phát triển một ứng dụng đa tầng.



Hình 8.7: Kiến trúc đa tầng.

Trong một ứng dụng đa tầng, lõi xử lý tính toán cho báo biểu tập trung trên một máy chủ chuyên dụng (Application Server). Crystal Reports Server và BusinessObjects Enterprise là bộ khung điều khiển mô hình ứng dụng này.

Các báo biểu tập trung trên một máy chủ chịu sự phân quyền, kiểm tra tính an toàn bảo mật, chia sẻ dữ liệu và khai thác sức mạnh tính toán của máy chủ. Ứng dụng Web hay những ứng dụng máy khách sẽ gửi yêu cầu (tham số, tên báo biểu ...) đến máy chủ. Máy chủ sẽ xử lý yêu cầu và trả về kết quả.

Tóm lại các ứng dụng Windows truyền thống thường sử dụng mô hình Thick-Client. Với môi trường ứng dụng Web, đa số những nhà phát triển sẽ chọn mô hình Single-Tier. Mô hình này có thể làm tăng giá thành khi phải mua thêm máy chủ và phần mềm bổ sung. Khi ứng dụng của bạn mở rộng trên quy mô nhiều người dùng, mô hình đa tầng Multitiered sẽ phù hợp hơn cả vì nó bảo đảm tính bảo mật, chia sẻ thông tin và khai thác sức mạnh xử lý của máy chủ.

5. TÓM TẮT

Mục đích của chương này là đưa ra các giải pháp để bạn chọn lựa khi thiết kế ứng dụng cần tích hợp báo biểu của Crystal. Không có mô hình hay kiến trúc nào tốt nhất cả, tùy theo mục đích và quy mô của ứng dụng mà bạn chọn giải pháp thích hợp. Trong các chương tiếp theo, chúng ta sẽ từng bước cụ thể hóa quá trình tích hợp báo biểu vào các loại ứng dụng Windows, .NET, Web và Java. Bắt đầu bằng những ứng dụng truyền thống Windows, chương sau sẽ hướng dẫn bạn cách sử dụng và liên kết báo biểu với ứng dụng .NET.

Chương 9:

TÍCH HỢP BÁO BIỂU VÀO ỨNG DỤNG WINDOWS

- ♦ Tổng quan.
- ♦ Làm việc với mô hình Report View Object.
- ♦ Làm việc với mô hình Report Document Object.
- ♦ Tóm tắt.

1. TỔNG QUAN

Trong những năm qua các ứng dụng của Window hầu như thống trị trên thị trường phần mềm máy tính, chúng bao gồm các ứng cá nhân, ứng dụng doanh nghiệp từ trình xử lý văn bản, bảng tính, cơ sở dữ liệu, quản lý khách hàng, tài chính, kế toán, thống kê...

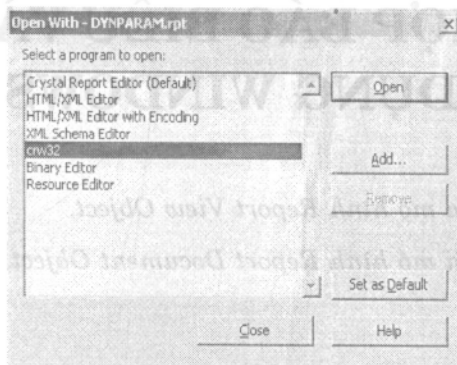
Hầu hết các ứng dụng doanh nghiệp đều cần quản lý một số lượng lớn dữ liệu. Một yêu cầu chung cho những kiểu ứng dụng này là khả năng trình bày thông tin sáng sủa, ngắn gọn - đây cũng là mục tiêu mà các chương trình cần đến Crystal Reports.

Một trong những công ty nhận ra nhu cầu sử dụng công cụ Crystal Reports là Microsoft. Crystal Reports được phát triển trong Microsoft từ Crystal Reports 2.0 và được gắn vào Visual Basic 3.0 từ năm 1993. Trên mỗi phiên bản, Crystal Reports được phát triển song hành cùng với môi trường lập trình ứng dụng của Microsoft, phiên bản mới nhất mà Microsoft tích hợp đầy đủ các chức năng của Crystal Reports là Visual Studio 2005.

Chương này sẽ hướng dẫn bạn cách tích hợp Crystal Reports vào trong những ứng dụng của Window được tạo ra từ môi trường Visual Studio .NET và .NET Framework.

Giả thiết rằng trong chương này bạn đã cài đặt Visual Studio .NET 2003 (hoặc Visual Studio 2005) cùng Crystal Reports XI Developers Edition. Bạn có thể soạn thảo các báo biểu nào trực tiếp trong môi trường Visual Studio IDE. Nếu thích sử dụng môi trường Crystal riêng để

thiết kế báo biểu độc lập bạn click phải vào báo biểu trong dự án và chọn Open With để mở hộp thoại như hình 9.1. Chọn **Crw32**. Đây là chương trình Crystal Reports XI.



Hình 9.1: Tùy chọn Open With.

Tất cả các ứng dụng mẫu và báo biểu của chương này được chứa trong CD ROM.

Với Crystal Reports XI, bạn có thể sử dụng hai mô hình đối tượng chính để tích hợp báo biểu vào trong ứng dụng .NET. Mô hình thứ nhất là mô hình đối tượng **CrystalReportViewer**. Mô hình này cung cấp các thuộc tính, phương thức và sự kiện để xem báo biểu và thực hiện các theo yêu cầu đơn giản của khách hàng như duyệt trang, in ấn. Mô hình thứ hai là mô hình đối tượng **ReportDocument**. Mô hình này cung cấp một tập hợp các thuộc tính, phương thức và sự kiện mà bạn có thể sử dụng để làm việc trực tiếp với báo biểu.

Quyết định sử dụng mô hình đối tượng nào là tùy vào kiểu ứng dụng mà bạn muốn tạo. Trong chương 8, chúng ta đã xem qua những phương pháp tích hợp khác nhau. Với những ví dụ đơn giản, mô hình đối tượng **CrystalReportViewer** là chọn lựa tốt nhất do nó cung cấp các tùy chọn mà bạn có thể xem trước (Preview), in và lưu dữ liệu báo biểu thành file... Bạn cũng có thể chuyển tham số cho báo biểu thông qua mô hình đối tượng này cũng như đăng nhập tự động vào cơ sở dữ liệu.

Nếu cần tích hợp sâu hơn, chẳng hạn chương trình cần chuyển tập dữ liệu đặc thù cho báo biểu qua ADO hoặc ADO.NET, in kết quả ra một máy in riêng biệt nào đó, điều khiển Field công thức và định dạng báo biểu theo cách của bạn, khi đó bạn nên sử dụng mô hình đối tượng **ReportDocument**.

2. LÀM VIỆC VỚI MÔ HÌNH REPORT VIEW OBJECT

Để sử dụng mô hình đối tượng *CrystalReportViewer* bạn cần làm hai việc. Đầu tiên là từ cửa sổ công cụ điều khiển .NET, bạn kéo và thả đối tượng *CrystalReportViewer* vào trong cửa sổ Form của ứng dụng; công việc thứ hai là viết mã cho Viewer thông qua các thuộc tính, phương thức và sự kiện liên quan đến mô hình đối tượng để điều khiển báo biểu và Viewer.

2.1. Xem trước báo biểu

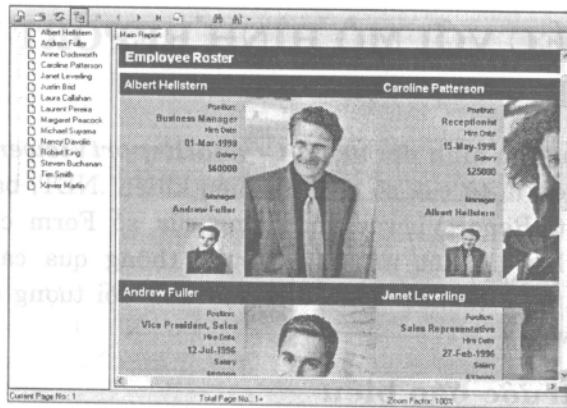
Bạn chỉ cần một dòng mã duy nhất để xem báo biểu từ ứng dụng. Để xem báo biểu trong ứng dụng, tất cả những gì bạn cần là kéo *CrystalReportViewer* lên trên form và đặt thuộc tính *ReportSource* trở vào file báo biểu mà bạn thiết kế trong Crystal hay trong dự án .NET. Dưới đây là các bước để hiển thị một báo biểu trong ứng dụng .NET:

1. Mở Visual Studio.NET 2003 (hoặc Visual Studio 2005).
2. Click File | New | Project.
3. Từ cửa sổ New Project, click vào mục Visual Studio Project, và sau đó chọn Windows Application.
4. Nhập tên dự án, click OK. Trong ví dụ này, chúng ta đặt tên dự án là VB_ViewReport. Một dự án mới được mở ra cùng với cửa sổ form mẫu.
5. Click View | Toolbox và kéo *CrystalReportViewer* từ cửa sổ ToolBox vào form, đặt thuộc tính Dock của *CrystalReportViewer* là Fill.
6. Tiếp theo, Double click vào form để viết mã cho sự kiện Form Load như sau:

```
CrystalReportViewer1.ReportSource="C:\Roster.rpt"
```

Báo biểu Roster.rpt được chứa trong thư mục Project trên CD ROM. Bạn có thể sửa đổi vị trí của dự án phụ thuộc vào nơi bạn lưu trữ file này.

7. Click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Cửa sổ Form sẽ hiển thị cùng với điều khiển *CrystalReportViewer* mà bạn đã thêm vào Form như hình 9.2.



Hình 9.2: Viewer hiển thị trong cửa sổ Form.

Theo mặc định, điều khiển Report Viewer được hiển thị với một thanh công cụ ở đỉnh bao gồm các nút sau:

- **Export:** Khi click vào nút **Export**, xuất hiện hộp thoại Save, bạn có thể kết xuất báo biểu sang định dạng Word, Excel, PDF, v.v...
- **Print:** Mở hộp thoại in.
- **Refresh:** Làm mới lại cơ sở dữ liệu.
- **Group Tree:** Hiển thị hoặc ẩn cây phân nhóm phía bên trái báo biểu.
- **Page Navigation Buttons:** Cung cấp các nút di chuyển, duyệt trang.
- **Find:** Mở hộp thoại Find nhập văn bản để tìm kiếm thông tin trong báo biểu.
- **Zoom:** Điều khiển phóng to, thu nhỏ nội dung báo biểu.

2.2. Sửa đổi diện mạo công cụ Viewer

Công cụ Viewer cung cấp một số thuộc tính cho phép bạn thay đổi diện mạo của nó.

Thuộc tính	Ý nghĩa	Kiểu
DisplayToolBar	Hiển thị thanh công cụ Viewer.	Boolean
DisplayBackGroundEdge	Hiển thị khung viền quanh Viewer.	Boolean

DisplaygroupTree	Hiển thị cấu trúc cây phân nhóm phía trái khung Viewer.	Boolean
------------------	---	---------

Ví dụ nếu muốn ẩn cây phân nhóm hoặc thanh công cụ, bạn có thể thêm đoạn mã sau vào Form Load:

```
CrystalReportViewer1.DisplayGroupTree=False
CrystalReportViewer1.DisplayToolBar=False
```

Nếu sử dụng mã này để tắt cây phân nhóm và thanh công cụ, báo biểu của bạn bây giờ sẽ trông như hình 9.3.



Hình 9.3: Report Viewer loại bỏ cây phân nhóm và thanh công cụ.

Bạn cũng có thể điều khiển các thành phần riêng lẻ trên thanh công cụ. Ví dụ, bạn có thể không muốn hiển thị nút in, khi đó người sử dụng sẽ không thể in nội dung báo biểu từ công cụ Viewer. Những thuộc tính sau có thể dùng để điều khiển các nút trên thanh công cụ:

- ❖ ShowCloseButton.
- ❖ ShowExportButton.
- ❖ ShowGotoPageButton.
- ❖ ShowGroupTreeButton.
- ❖ ShowPageNavigationButtons.

❖ ShowPrintButton.

❖ ShowRefreshButton.

❖ ShowTextSearchButton.

Tất cả các thuộc tính là Boolean, để dấu nút in bạn có thể viết mã như sau:

```
CrystalReportViewer1.ShowPrintButton=False
```

Khi xem báo biểu nút in Print sẽ không còn xuất hiện trên công cụ Viewer nữa (hình 9.4).



Hình 9.4: Báo biểu công cụ xem thiếu nút Print.

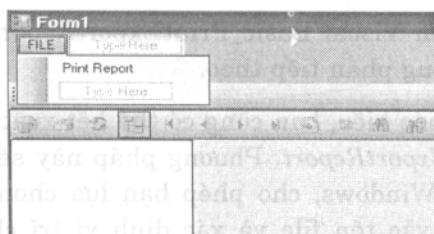
2.3. In và kết xuất dữ liệu

Mặc dù chính Report Viewer cung cấp nút in Print để người sử dụng có thể click vào đó in báo biểu, nhưng bạn có thể chủ động in bằng cách gọi phương thức PrintReport của Viewer. Mục in lúc này có thể được tùy biến để trong menu chính.

Trong ví dụ này, chúng ta sẽ thêm Report Viewer vào trong form, sau đó tạo một mục menu tên là Print Report gắn trên menu File của chương trình. Khi người sử dụng click File | Print Report, một hộp thoại in chuẩn của Windows sẽ xuất hiện và cho phép người sử dụng chọn để in nội dung báo biểu. Để tạo ứng dụng này, thực hiện các bước sau:

1. Click Start | Visual Studio.NET 2003.
2. Click File | New | Project.
3. Từ cửa sổ New Project, click vào mục Visual Studio Project và sau đó chọn Windows Application.
4. Nhập tên cho dự án và click OK. Trong ví dụ này, chúng ta đặt tên cho dự án là VB_ViewReport.
5. Khi dự án mới xuất hiện với cửa sổ form trống, click View | Toolbox và chọn công cụ điều khiển MainMenu từ cửa sổ ToolBox.

Kéo công cụ điều khiển MainMenu vào trong Form. Trong ô Type Here, gõ vào: File. Một ô xuất hiện ngay bên dưới, gõ vào: Print Report, như hình 9.5.



Hình 9.5: Thêm menu vào cửa sổ chương trình.

6. Tiếp theo, chọn thành phần điều khiển *CrystalReportViewer* trong cửa sổ công cụ và kéo nó vào Form. Đặt thuộc tính **Dock** cho *CrystalReportViewer* là **Fill**.

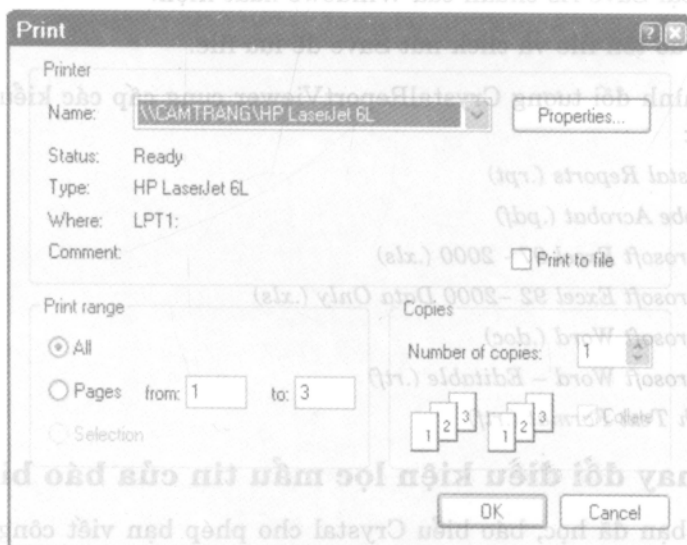
7. Bạn viết mã cho sự kiện Form Load nạp báo biểu như sau:

```
CrystalReportViewer1.ReportSource="d:\roster.rpt"
```

8. Trên Form, double click vào tùy chọn Print Report trên menu chính và nhập vào đoạn mã sau để hiển thị hộp thoại in báo biểu:

```
CrystalReport Viewer1.PrintReport
```

9. Click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Form sẽ hiển thị báo biểu, khi click File | Print Report, hộp thoại in chuẩn của Windows xuất hiện (hình 9.6).



Hình 9.6: Màn hình PrintReport.

10. Lưu đề án với tên Visual Basic_PrintExportReport; chúng ta sẽ tiếp tục đề án này trong phần tiếp theo.

Ngoài việc in báo biểu, bạn cũng có thể kết xuất số liệu báo biểu sử dụng phương pháp *ExportReport*. Phương pháp này sẽ mở một hộp thoại Save As chuẩn của Windows, cho phép bạn lựa chọn dạng kết xuất dữ liệu cũng như nhập vào tên file và xác định vị trí cho file kết xuất dữ liệu. Để lưu dữ liệu của báo biểu sang định dạng khác, bạn thực hiện các bước sau:

1. Click Start | Visual Studio.NET 2003 (hoặc Visual Studio 2005).
2. Click File | Open | Project. Mở dự án Visual Basic_PrintExportReport mà bạn đang làm việc trong phần trước.
3. Dưới tùy chọn Print Report, trong hộp thoại Type Here gõ vào một mục menu mới mang tên Export Report.
4. Tiếp theo, double click tùy chọn Export Report để viết mã cho mục chọn trên menu này như sau:

```
CrystalReportViewer1.ExportReport()
```

5. Click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Form xuất hiện và trình bày báo biểu. Khi click File | Export Report, hộp thoại Save As chuẩn của Windows xuất hiện.
6. Nhập vào tên file và click nút Save để lưu file.

Mô hình đối tượng CrystalReportViewer cung cấp các kiểu kết xuất dữ liệu sau:

Crystal Reports (.rpt)

Adobe Acrobat (.pdf)

Microsoft Excel 97 - 2000 (.xls)

Microsoft Excel 92 - 2000 Data Only (.xls)

Microsoft Word (.doc)

Microsoft Word - Editable (.rtf)

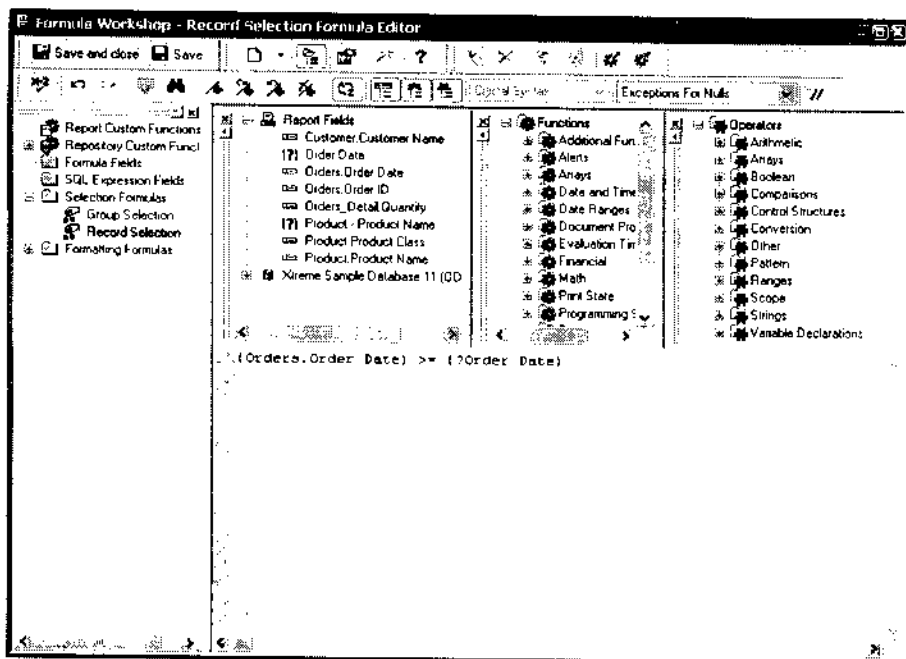
Rich Text Format (.rtf)

2.4. Thay đổi điều kiện lọc mẫu tin của báo biểu

Như bạn đã học, báo biểu Crystal cho phép bạn viết công thức lọc mẫu tin thông qua Record Selection. Ví dụ:

```
{Customer.Country}="USA"
```

Công thức lọc mẫu tin trên sẽ lọc và hiển thị những mẫu tin có mã quốc gia Country là USA. Bạn có thể xem các điều kiện lọc đang sử dụng trong báo biểu bằng cách chọn Report | Selection Formulas | Record trong môi trường Crystal Reports. Hộp thoại Record Selection Formula Editor sẽ xuất hiện như hình 9.7.



Hình 9.7: Hộp thoại Record Selection Formula Editor.

Bất kỳ điều kiện lọc mẫu tin nào mà bạn tạo ra trong công cụ này đều lưu trữ cùng với báo biểu. Khi tích hợp các báo biểu vào trong ứng dụng, bạn có hai thuộc tính khác nhau để đặt điều kiện lọc cho mẫu tin. Thuộc tính thứ nhất, *SelectionFormula*, có thể được sử dụng để tìm hoặc đặt công thức lọc mẫu tin cho báo biểu. Thuộc tính thứ hai, *ViewTimeSelectionFormula*, được sử dụng để kết nối với một công thức lọc báo biểu dựa trên công thức lọc đã tồn tại. Ví dụ, nếu báo biểu đã có một điều kiện lọc mẫu tin:

```
{Customer.Country}="USA"
```

bạn sử dụng thuộc tính *ViewTimeSelectionFormula* để thêm điều kiện lọc:

```
{Customer.Country}="NSW"
```

kết quả công thức lọc mẫu tin sẽ trở thành:

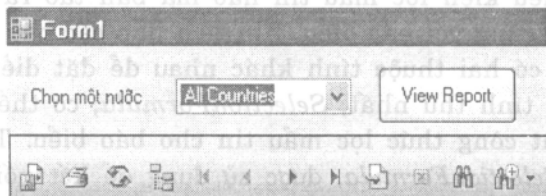
```
{Customer.Country}="USA" and {Customer.Country}="NSW"
```

Ví dụ dưới đây sẽ hướng dẫn bạn cách thiết lập công thức lọc mẫu tin khi đưa báo biểu vào ứng dụng:

1. Từ menu Start, mở Visual Studio .NET 2003 (hoặc Visual Studio 2005).
2. Click File | New | Project.
3. Từ cửa sổ New Project, click mục Visual Studio Project và chọn Windows Application.
4. Nhập vào tên dự án, click OK. Trong ví dụ này, chúng ta đặt tên dự án là VB_RecordSelection.
5. Một dự án mới xuất hiện cùng cửa sổ Form trống. Click View | Toolbox và kéo thành phần điều khiển *CrystalReportViewer* vào trong Form. Đặt thuộc tính *Dock* của *CrystalReportViewer* là *Fill*.
6. Viết mã cho sự kiện Form Load để nạp báo biểu Dynselect.RPT. Báo biểu mẫu Dynselect.RPT được chứa trong thư mục Project trên CD ROM. Bạn có thể sửa đổi vị trí file cho phù hợp với đường dẫn.

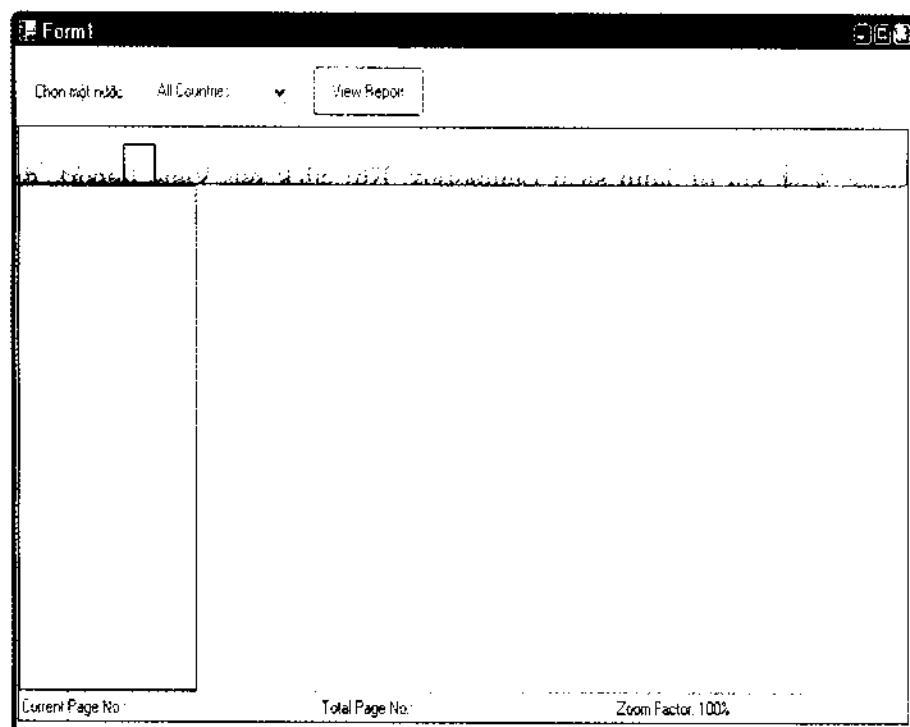
```
CrystalReportViewer1.ReportSource= "d ; \dynselect.rpt"  
CrystalReportViewer1.DisplayGroupTree=False
```

7. Tiếp theo, bạn kéo thêm vào Form một đối tượng nhãn Label, một đối tượng ComboBox, và một đối tượng Button đặt phía trên Viewer như hình 9.8.



Hình 9.8: Các thành phần dùng để lọc dữ liệu báo biểu.

8. Thay đổi thuộc tính Text của Label là *Select a Country*. Và của Combo Box là *All Countries*, của Button là *View Report*. Giao diện Form bây giờ sẽ giống như hình 9.9.



Hình 9.9: Form được hoàn thành.

9. Bây giờ chúng ta cần nhập vào danh sách một số nước để người dùng lựa chọn. Bạn chọn thuộc tính Items của ComboBox và nhập vào các tên nước sau:
- ◆ USA
 - ◆ Australia
 - ◆ Canada
 - ◆ Mexico
10. Tiếp theo chúng ta cần viết mã cho nút lệnh View Report để tạo công thức lọc mẫu tin và chuyển nó vào trong báo biểu. Bạn Double click vào nút View Report để viết mã cho sự kiện Click của nút nhấn như sau:

```
If ComboBox1.Text ="All Countries" Then
    Crystal ReportsViewer1.SelectionFormula=""
Else
    CrystalReportViewer1.SelectionFormula="
        {Customer.Country} "= " & ComboBox1.Text & "
```

End if

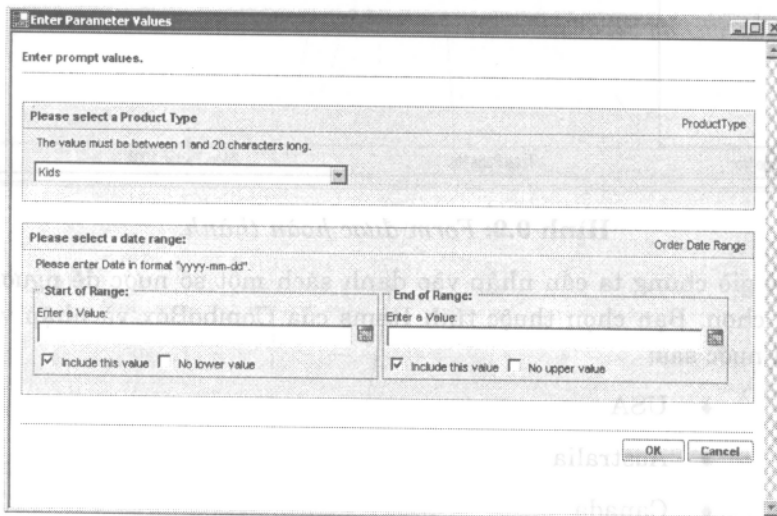
Crystal ReportsViewer1.RefreshReport ()

11. Cuối cùng, click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Form xuất hiện cùng với nội dung báo biểu. Bạn có thể chọn các quốc gia từ danh sách ComboBox. Khi click nút View Report, điều kiện lọc mẫu tin sẽ được áp dụng và báo biểu sẽ được cập nhật lại.

Đây là một ví dụ đơn giản áp dụng điều kiện lọc mẫu tin cho báo biểu. Bạn có thể tạo ra các công thức lọc phức tạp hơn theo yêu cầu của mình.

2.5. Làm việc với các Field tham số

Với báo biểu trong Crystal Reports, bạn có thể truyền tham số động và tham số tĩnh. Khi thực thi báo biểu, Crystal sẽ hiển thị hộp thoại để bạn chọn hay nhập vào các giá trị cho tham số (hình 9.10).



Hình 9.10: Hộp thoại truyền tham số trong Crystal Reports XI.

Tuy nhiên, nếu muốn tùy biến hộp thoại nhập giá trị tham số theo cách riêng của ứng dụng, bạn sẽ chuyển giá trị tham số cho Report Viewer thông qua lớp *ParameterField*. Ngoài ra, còn một lớp khác mang tên *ParameterFields* (chú ý số nhiều có 's'), là tập hợp chứa tất cả các Field tham số trong báo biểu.

Để làm việc với các Field tham số, trước hết chúng ta sẽ xem làm cách nào có thể đọc các thông tin về Field tham số trong báo biểu.

2.5.1. Đọc thông tin từ Field tham số

Ngoài việc đặt giá trị cho Field tham số, bạn có thể đọc thông tin về các Field tham số trong báo biểu. Những thông tin cơ bản về Field tham số gồm:

- *Name*: Tên của tham số.
- *ReportParametersType*: Kiểu của tham số (String, Date, v.v...).
- *ParameterValueType*: Kiểu giá trị được yêu cầu bởi tham số (String, Date, v.v...).
- *PromptText*: Văn bản được hiển thị nhắc nhở người sử dụng nhập vào giá trị cho tham số.
- *ReportName*: Tên của báo biểu chính hoặc báo biểu con nơi tham số xuất hiện. Nếu nó xuất hiện trong phần chính của Report thì thuộc tính này sẽ được để trống.
- *HasCurrentValue*: Cho biết tham số này có một giá trị hay không.
- *MinimumValue*: Trả về giá trị nhỏ nhất sẽ được tham số chấp nhận.
- *MaximumValue*: Trả về giá trị lớn nhất sẽ được tham số chấp nhận.
- *EditMask*: Mặt nạ nhập liệu cho tham số.

Để tìm hiểu các thuộc tính này làm việc như thế nào, trong ví dụ tiếp theo của chúng ta sẽ lần lượt sử dụng chúng:

1. Mở Visual Studio .NET 2003 (hoặc Visual Studio 2005).
2. Click File | New | Project.
3. Từ cửa sổ New Project, click vào mục Visual Studio Project và chọn Windows Application.
4. Nhập tên và xác định đường dẫn chứa dự án và click OK. Trong ví dụ này chúng ta đặt tên dự án là VB_Parameters.
5. Một dự án mới sẽ xuất hiện cùng cửa sổ Form trống. Click View | ToolBox và kéo thành phần điều khiển *CrystalReportViewer* vào Form. Đặt thuộc tính Dock cho *CrystalReportViewer* là Fill.
6. Viết mã cho sự kiện Form Load để nạp báo biểu Paraminfo.RPT. Báo biểu mẫu Paraminfo.RPT được chứa trong thư mục Project trên CD ROM. Bạn có thể sửa đổi vị trí file cho phù hợp với đường dẫn.

```
CrystalReportViewer1.ReportSource="d:\Paraminfo.rpt"
CrystalReportViewer1.DisplayGroupTree=False
```

- Để kiểm tra báo biểu nạp đúng đắn, click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Form xuất hiện và hiển thị báo biểu. Đóng cửa sổ quay trở về môi trường Visual Studio.
- Tiếp theo chúng ta có thể khai báo các biến tham số *ParameterField*. Bạn viết thêm đoạn mã sau cho sự kiện Form Load:

```
CrystalReportViewer1.ReportSource="d:\Paraminfo.rpt"
CrystalReportViewer1.DisplayGroupTree=False
```

'Phần mã viết thêm

```
Dim ParameterFields As
    CrystalDecisions.Shared.ParameterFields
Dim ParameterField As
    CrystalDecisions.Shared.ParameterField

ParameterFields =
    CrystalReportViewer1.ParameterFieldInfo

Dim DisplayText
For Each ParameterField In ParameterFields
    DisplayText = " Parameter Name: " &
        ParameterField.Name & Chr(13)
    DisplayText = DisplayText & " Parameter Type: " &
        ParameterField.ReportParameterType.ToString
        & Chr(13)
    DisplayText = DisplayText &
        " Parameter Value Type: " &
        ParameterField.ParameterValueType.ToString &
        Chr(13)
    DisplayText = DisplayText & " Prompting Text: " &
        ParameterField.PromptText & Chr(13)
    DisplayText = DisplayText & " Report Name: " &
        ParameterField.ReportName & Chr(13)
    DisplayText = DisplayText & " Has Current Value: " &
        ParameterField.HasCurrentValue & Chr(13)
    DisplayText = DisplayText & " Minimum Value: " &
        ParameterField.MinimumValue & Chr(13)
    DisplayText = DisplayText & " Maximum Value: " &
        ParameterField.MaximumValue & Chr(13)
    DisplayText = DisplayText & " Edit Mask: " &
        ParameterField.EditMask & Chr(13)
    MsgBox(DisplayText)
Next
```

9. Click Debug | Start hoặc nhấn phím F5 để thực thi lại ứng dụng. Một hộp thoại thông báo sẽ được hiển thị cho biết thông tin về các Field tham số lưu trong báo biểu.
10. Click OK để tiếp tục, nhập vào giá trị cho tham số và click OK để xem báo biểu. Lưu dự án với tên VB_Parameter.

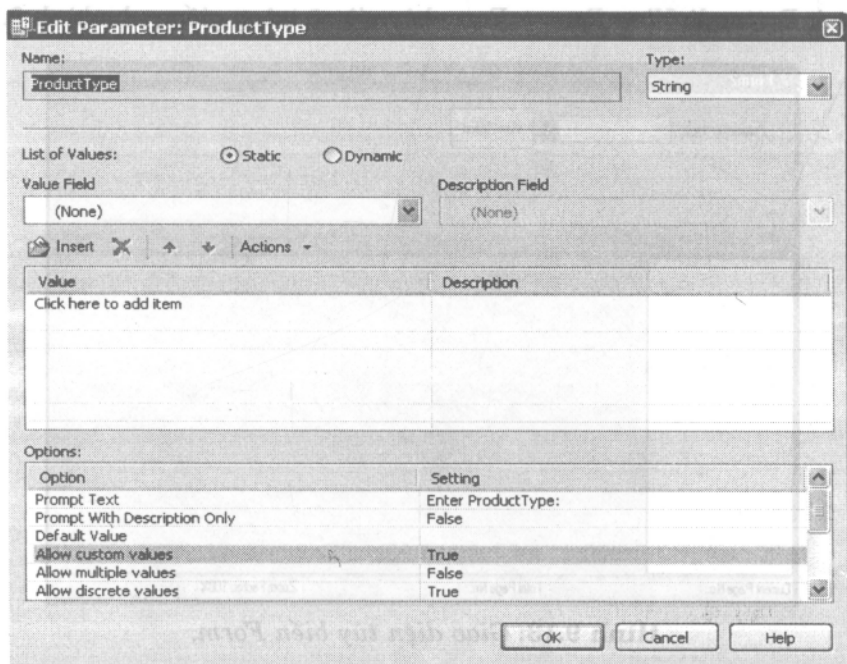
Trong phần tiếp theo chúng ta sẽ học cách đọc các giá trị mặc định của tham số lưu trong báo biểu và gán giá trị mới cho Field tham số.

2.5.2. Đọc và đặt các giá trị mới cho field tham số

Khi tạo các Field tham số trong Crystal Reports, bạn có thể tạo một danh sách giá trị tĩnh cho phép người sử dụng chọn. Danh sách này được gọi là *giá trị mặc định* cho tham số.

Ví dụ, nếu có một tham số State, bạn có thể tạo giá trị mặc định bao gồm chuỗi diễn giải New York và một giá trị kèm theo là NY. Hình 9.11 là cửa sổ tạo các giá trị mặc định cho tham số trong Crystal Reports.

Khi người sử dụng được yêu cầu chọn hay nhập giá trị cho tham số, danh sách các giá trị mặc định này sẽ hiển thị như hình 9.12.



Edit Parameter: ProductType

Name: Type:

List of Values: ☒ Static ☐ Dynamic

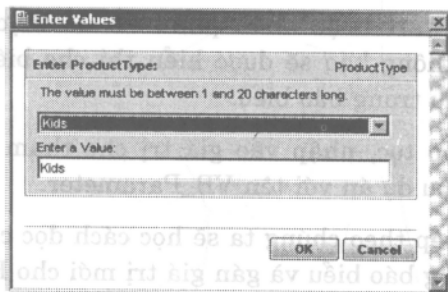
Value Field: Description Field:

Value	Description
Click here to add item	

Options:

Option	Setting
Prompt Text	Enter ProductType:
Prompt With Description Only	False
Default Value	
Allow custom values	True
Allow multiple values	False
Allow discrete values	True

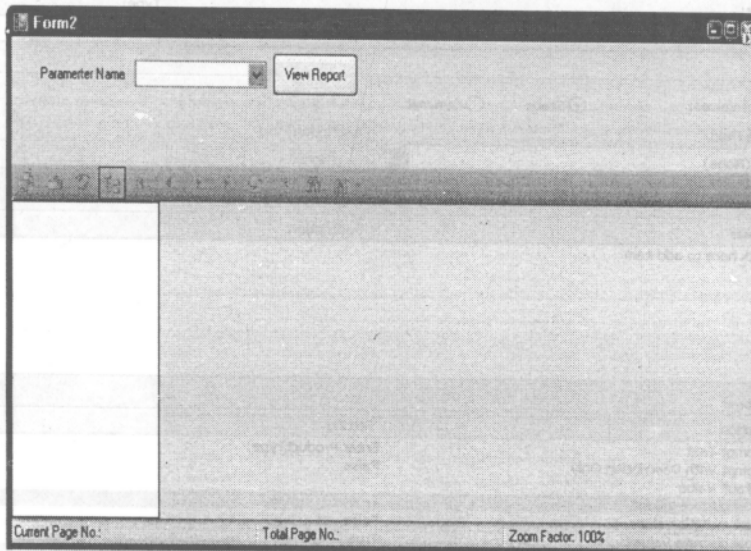
Hình 9.11: Đặt các giá trị mặc định cho tham số.



Hình 9.12: Hiện hành và nhiều giá trị mặc định.

Để đọc giá trị mặc định của tham số bạn sử dụng thuộc tính *DefaultValues*. Trong dự án ví dụ tiếp theo sau, chúng ta sẽ đọc các giá trị mặc định của tham số và điền chúng vào ComboBox, người sử dụng có thể chọn các giá trị trong ô ComboBox này và click vào một nút để thực thi báo biểu. Chương trình sẽ chuyển giá trị tham số cho báo biểu bằng mã lệnh.

1. Mở dự án VB_Parameters chúng ta đang làm việc trong mục trước.
2. Thêm một đối tượng nhãn Label, một đối tượng ComboBox và một nút nhấn Button vào Form. Gán thuộc tính Text của ComboBox là "", của nút Button là *View Report*. Form bây giờ sẽ trông giống như hình 9.13.



Hình 9.13: Giao diện tùy biến Form.

3. Double click vào Form để mở sự kiện Form Load, bạn thay đoạn mã trước đó bằng đoạn mã sau:

```
CrystalReportViewer1.ReportSource="d: \Dynparm.rpt"  
CrystalReportViewer1.DisplayGroupTree=False  
ParameterFields =
```

```
    CrystalReportViewer1.ParameterFieldInfo
```

```
ParameterField =ParameterFields(0)  
    Label1.Text= ParameterField.PromptText
```

```
For I As Integer= 0 to
```

```
    (ParameterField.DafaultValues.Count-1 )  
    ComboBox1.Item.Add(ParameterField.
```

```
    DefaultValues.Item(i). Description)
```

```
Next
```

```
ParameterDiscreteValue=New  
CrystalDecisions.Shared.ParameterDiscreteValue  
ParameterDiscreteValue.Value=""
```

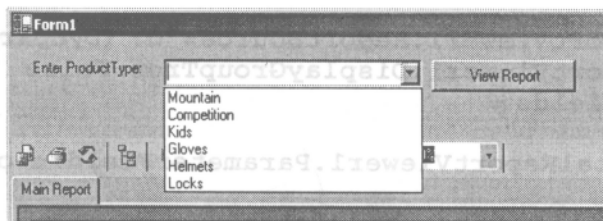
```
ParameterField.CurrentValues.Add(  
    ParameterDiscreteValue)
```

4. Double click vào nút View Report trong Form và nhập đoạn mã như sau cho sự kiện Click của nút nhấn:

```
CrystalReportViewer1.ReportSource="d: \ Dynparm.rpt"  
ParameterFields = CrystalReportViewer1.Parameter  
FieldInfoParameterField =ParameterFields(0)
```

```
    ParameterDiscreteValue=New  
CrystalDecisions.Shared.ParameterDiscreteValue  
ParameterDiscreteValue.Value=ComboBox1.SelectedItem  
ParameterField.CurrentValues.Add(  
    ParameterDiscreteValue)
```

5. Click Debug |Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Click vào ComboBox để chọn giá trị cho tham số định nghĩa sẵn trong báo biểu (hình 9.14).
6. Chọn một giá trị và click vào nút View Report. Bây giờ báo biểu sẽ được cập nhật lại theo giá trị tham số mà bạn đã chọn.



Hình 9.14: Form với danh sách giá trị mặc định đọc được từ báo biểu.

7. Thay vì đọc các giá trị mặc định từ báo biểu, bạn có thể tự xây dựng các giá trị mặc định này trong ứng dụng của mình, sau đó gọi phương thức là `CurrentValues.Add` để chuyển giá trị tham số cho báo biểu thực thi.

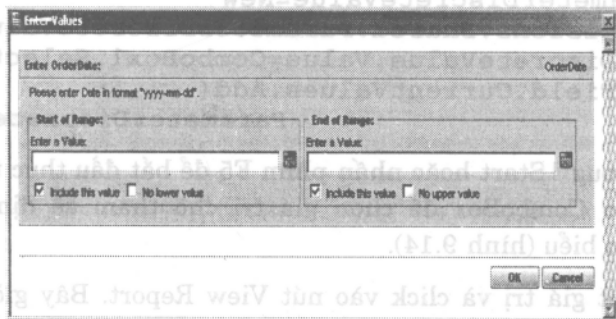
Ghi chú: Một thủ thuật khác, khi nào làm việc với báo biểu mà báo biểu đó có chứa báo biểu con, trong báo biểu con yêu cầu truyền tham số, bạn có thể chuyển giá trị tham số cho báo biểu con như sau:

```
ParameterField = ParameterField(0, "SubReport1")
```

2.5.3. Thiết đặt giá trị phạm vi cho Field tham số

Một kiểu Field tham số phổ biến khác là tham số phạm vi. Tham số phạm vi được định nghĩa bởi một giá trị cận trên và cận dưới. Field tham số phạm vi thường được sử dụng với kiểu Date hoặc kiểu Number.

Ví dụ, bạn có thể tạo ra một tham số phạm vi cho phép bạn chọn những khách hàng với doanh số đạt được trong khoảng \$10000 đến \$20000. Hoặc bạn có thể tạo ra một tham số phạm vi cho phép nhập vào ngày bắt đầu và ngày kết thúc, như vậy bạn có thể chọn tất cả mẫu tin nằm trong khoảng giữa của hai ngày này (hình 9.15).



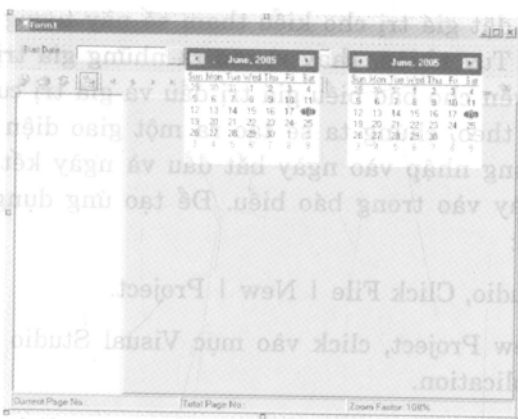
Hình 9.15: Tham số phạm vi theo kiểu ngày tháng gồm ngày bắt đầu và ngày kết thúc.

Bạn có thể đặt giá trị cho kiểu tham số này tương tự những tham số thông thường. Tuy nhiên, thay vì tạo ra những giá trị riêng biệt đơn giản, bạn sẽ chuyển cho báo biểu giá trị đầu và giá trị cuối của tham số. Trong ví dụ tiếp theo, chúng ta sẽ tạo ra một giao diện thân thuộc cho phép người sử dụng nhập vào ngày bắt đầu và ngày kết thúc và sau đó chuyển giá trị này vào trong báo biểu. Để tạo ứng dụng này, bạn thực hiện các bước sau:

1. Mở Visual Studio, Click File | New | Project.
2. Từ cửa sổ New Project, click vào mục Visual Studio Project và chọn Windows Application.
3. Nhập tên và xác định đường dẫn chứa dự án và click OK. Trong ví dụ này chúng ta đặt tên dự án là VB_RangeParam.
4. Một dự án mới sẽ được mở cùng với cửa sổ Form trống. Click View | ToolBox và kéo điều khiển *CrystalReportViewer* vào Form. Đặt thuộc tính Dock cho *CrystalReportViewer* là Fill.
5. Viết mã cho sự kiện Form Load để nạp báo biểu Rangeparam.RPT. Báo biểu mẫu Rangeparam.RPT được chứa trong thư mục Project trên CD ROM. Bạn có thể sửa đổi vị trí file cho phù hợp với đường dẫn.

```
CrystalReportViewer1.ReportSouce="d:\ RangeParam.rpt"  
CrystalReportViewer1.DisplayGroupTree=False
```

6. Click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Form xuất hiện cùng với nội dung báo biểu. Đóng cửa sổ Form và trở về cửa sổ Visual Studio.
7. Tiếp theo, thêm hai TextBox và ba nút Button vào Form. Gán thuộc tính Text cho 2 TextBox là "" và thuộc tính Text của hai nút Button là "...". Nút Button thứ ba đặt tên là View Report. Thêm hai nhãn Label trước hai TextBox là Start Date và End Date. Bây giờ giao diện Form sẽ trông giống như hình 9.16.
8. Để chọn ngày tháng một cách dễ dàng, bạn kéo điều khiển *MonthCalendar* trên cửa sổ công cụ ToolBox đặt gần các nút nhấn. Khi người dùng kích vào nút nhấn, đối tượng lịch này sẽ hiển thị cho phép bạn chọn ngày tháng.
9. Thay đổi thuộc tính Visible của hai điều khiển *MonthCalendar* là False.



Hình 9.16: Giao diện Form hoàn tất.

10. Tiếp theo, nhập đoạn mã sau cho sự kiện Click của nút nhấn:

```
MonthCalendar1.Show()
```

11. Nhấp đôi chuột vào điều khiển *MonthCalendar* để mở sự kiện *DateSelected* khi người sử dụng chọn lựa ngày tháng. Trong sự kiện này chúng ta sẽ chuyển giá trị ngày tháng được chọn cho ô *TextBox*.

```
Private Sub MonthCalendar1_DateSelected(  
    ByVal sender As System.Object,  
    ByVal e As DateRangeEventArgs)  
    Handles MonthCalendar1.DateSelected  
    TextBox1.Text = MonthCalendar1.DateSelected  
    MonthCalendar1.Hide()  
End Sub
```

12. Tiếp theo bạn khai báo và nhập vào đoạn mã dưới đây trong lớp Form:

```
Dim ParameterFields As  
    CrystalDecisions.Shared.ParameterFields  
Dim ParameterField As  
    CrystalDecisions.Shared.ParameterField  
Dim ParameterRangeValue As  
    CrystalDecisions.Shared.ParameterRangeValues
```

13. Mở lại sự kiện Form Load, viết mã cho sự kiện này để thiết đặt giá trị Field tham số cho báo biểu.

```
CrystalReportViewer1.  
    ReportSource="d:\ RangeParm.rpt"  
CrystalReportViewer1.DisplayGroupTree=False
```

```
ParameterFields =  
    CrystalReportViewer1.ParameterFieldInfo  
  
ParameterField =ParameterFields(0)  
ParameterRangeValue =New  
CrystalDecision.Shared.ParameterRangeValue  
ParameterRangeValue.StartValue="1/1/1"  
ParameterRangeValue.EndValue="1/1/1"  
  
ParameterRangeValue.LowerBoundType=CrystalDecision.  
    [Shared].RangeBoundType.BoundInclusive  
ParameterRangeValue.UpperBoundType=CrystalDecision.  
    [Shared].RangeBoundType.BoundInclusive  
ParameterField.CurrentValues.Add(ParameterRangeValue)  
  
14. Double click vào nút View Report và nhập đoạn mã tiếp theo để thực  
    thi khi người sử dụng click nút nhấn. Đoạn mã này hầu như không  
    khác gì với đoạn mã sử dụng trong sự kiện Form Load.  
  
CrystalReportViewer1.ReportSource="d:\RangeParm.rpt"  
CrystalReportViewer1.DisplayGroupTree=False  
  
ParameterFields =  
    CrystalReportViewer1.ParameterFieldInfo  
ParameterField =ParameterFields(0)  
  
ParameterRangeValue =New  
    CrystalDecision.Shared.ParameterRangeValue  
  
ParameterRangeValue.StartValue=TextBox1.Text  
ParameterRangeValue.EndValue=TextBox2.Text  
  
ParameterRangeValue.LowerBoundType=CrystalDecision.  
    [Shared].RangeBoundType.BoundInclusive  
ParameterRangeValue.UpperBoundType=CrystalDecision.  
    [Shared].RangeBoundType.BoundInclusive  
ParameterField.CurrentValues.Add(ParameterRangeValue)
```

15. Click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng.
 Form xuất hiện, bạn có thể nhập vào ngày bắt đầu và ngày kết thúc.

Click vào nút (...) để mở khung lịch biểu cho phép bạn chọn ngày tháng thông qua MonthCalendar, click vào nút View Report để chuyển giá trị này cho báo biểu và xem nội dung báo biểu đã được lọc lại.

2.6. Đăng nhập cơ sở dữ liệu

Hầu hết các hệ cơ sở dữ liệu lớn đều có chế độ bảo mật yêu cầu bạn cung cấp thông tin đăng nhập trước khi cho phép báo biểu truy xuất nguồn dữ liệu. Thông thường khi bạn thực thi báo biểu, Crystal sẽ hiển thị hộp thoại yêu cầu bạn nhập vào tên và mật khẩu dùng cho đăng nhập vào cơ sở dữ liệu như hình 9.17. Để người dùng không phải nhập thông tin đăng nhập này mỗi khi thực thi báo biểu, bạn gán thông tin đăng nhập sẵn cho Viewer thông qua lớp LogonInfo như sau:

```
Dim LogonInfo as CrystalDecisions.Shared.TableLogOnInfo
```

```
CrystalReportViewer1.LogOnInfo=New
```

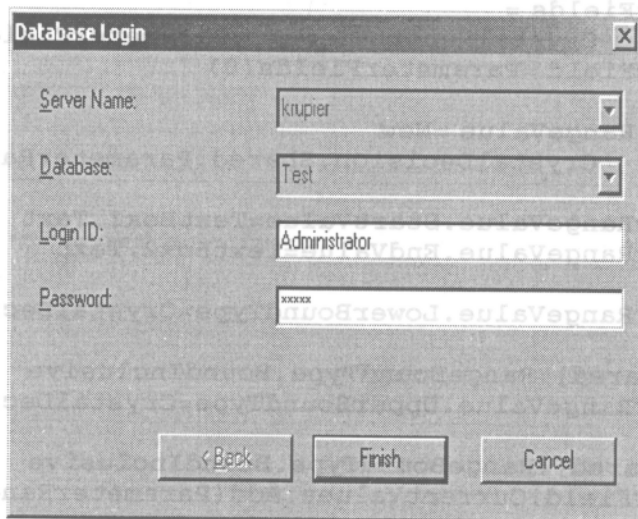
```
CrystalDecisions.Shared.TableLogOnInfos
```

```
LogOnInfo.TableName="Products"
```

```
LogOnInfo.ConnectionInfo.UserID="Admin"
```

```
LogOnInfo.ConnectionInfo.Password=""
```

```
CrystalReportViewer1.LogOnInfo.Add(LogOnInfo)
```



Hình 9.17: Nhập vào tên và mật khẩu để đăng nhập vào nguồn cơ sở dữ liệu.

Bạn phải chỉ định rõ bảng dữ liệu nào muốn truy xuất cùng với tên đăng nhập và mật khẩu. Bạn không thể áp dụng một lần thiết lập thông tin đăng nhập cho mọi bảng. Muốn thực hiện điều này bạn phải dùng mô hình đối tượng ReportDocument thay cho CrystalReportViewer. Mô hình đối tượng ReportDocument cho phép duyệt qua tất cả các bảng trong báo biểu và thiết lập các thông số đăng nhập cần thiết.

3. LÀM VIỆC VỚI MÔ HÌNH ĐỐI TƯỢNG REPORTDOCUMENT

Mô hình đối tượng ReportDocument cung cấp các phương thức và thuộc tính cho phép bạn điều khiển ở mức thấp tất cả thông tin có trong cấu trúc của báo biểu.

3.1. Chọn phương thức ràng buộc

Các báo biểu mà chúng ta làm việc cho đến bây giờ thường sử dụng mô hình đối tượng Viewer chỉ định file báo biểu thông qua thuộc tính ReportSource:

```
Crystal ReportsViewer1.ReportSource="c:\myreport.rpt"
```

Tuy nhiên, đây không phải là cách duy nhất mà bạn có thể dùng để truy cập báo biểu từ ứng dụng (mặc dù nó được xem là cách dễ nhất). Khi tích hợp Crystal Reports vào trong ứng dụng, bạn có thể chọn đặt báo biểu độc lập bên ngoài ứng dụng hoặc nhúng chúng vào trong cùng ứng dụng. Lợi ích của việc lưu báo biểu ngoài ứng dụng là ứng dụng có thể thêm nhiều báo biểu mới hoặc việc sửa đổi những báo biểu đã tồn tại. Nhúng báo biểu trực tiếp vào ứng dụng làm cho ứng dụng chạy nhanh hơn và không cho phép người dùng thay đổi những báo biểu quan trọng.

Để kết nối tới một báo biểu được lưu bên ngoài ứng dụng, bạn thực hiện các bước sau:

1. Mở Visual Studio, click chọn File | New | Project.
2. Từ cửa sổ New Project, click vào mục Visual Studio Project và chọn Windows Application.
3. Nhập tên cho dự án và click OK. Trong ví dụ này chúng ta đặt tên dự án là VB_ExternalBind. Một dự án mới sẽ xuất hiện cùng cửa sổ Form trống.

4. Bạn thêm dòng sau vào đầu lớp Form, khai báo sử dụng thư viện Crystal:

```
Imports Crystal Decisions.CrystalReport .Engine
```

5. Tiếp theo, kéo điều khiển *CrystalReportViewer* vào trong Form và đặt thuộc tính Dock là Fill.
6. Double click vào Form để viết mã cho sự kiện Form Load như sau:

```
Dim ExternalBindReport as ReportDocument =  
    New ReportDocument  
ExternalBindReport.Load ("C:\binding.rpt")  
CrystalReportViewer1.ReportSource=ExternalBindReport
```

7. Click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng, báo biểu sẽ hiển thị mặc dù chúng không nằm trong ứng dụng, bạn có thể đặt báo biểu ở bất kỳ đâu và chỉ cần chỉ đường dẫn để Crystal nạp báo biểu vào ứng dụng.

Để kết nối với một báo biểu lưu trữ bên trong ứng dụng, bạn thực hiện các bước sau:

1. Mở Visual Studio, click chọn File | New | Project.
2. Từ cửa sổ New Project, click vào mục Visual Studio Project và chọn Windows Application.
3. Nhập tên cho dự án và click OK. Trong ví dụ này chúng ta đặt tên dự án là VB_ EmbeddedBind. Một dự án mới sẽ xuất hiện cùng cửa sổ Form trống.
4. Đầu tiên chúng ta cần chỉ rõ báo biểu muốn làm việc, để thêm một báo biểu vào trong dự án, chọn Project | Add Existing Item. Bạn hãy chọn một báo biểu mẫu. Ví dụ ta chọn báo biểu binding.rpt chứa trong thư mục Projects trên đĩa CD ROM.
5. Từ thanh công cụ, kéo biểu tượng ReportDocument lên Form. Hộp thoại sẽ xuất hiện, nó sẽ cho phép bạn chọn các báo biểu hiện có trong dự án. Chọn một báo biểu và click OK.
6. Tiếp theo, viết mã cho sự kiện Form Load của Form như sau:

```
Dim EmbeddedBindReport as New Binding
```

7. Bây giờ bạn có thể làm việc và điều khiển báo biểu này thông qua thuộc tính, phương thức của EmbeddedBindReport trong mô hình đối

tượng *ReportDocument*. Để xem báo biểu, bạn kéo điều khiển *CrystalReportViewer* vào trong Form, tiếp theo nhập thêm đoạn mã sau vào Form Load:

```
CrystalReportViewer1.ReportSource= EmbeddedBindReport
```

8. Sau cùng, click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Báo biểu của bạn giờ đây được biên dịch và kết hợp vào chung với ứng dụng khi thực thi, Viewer sẽ đọc báo biểu từ bên trong ứng dụng và thực thi tương tự như trường hợp báo biểu được lưu bên ngoài. Bạn không cần chép file báo biểu theo chương trình khi cài đặt ứng dụng cho người dùng cuối.

3.2. In báo biểu

Mặc dù công cụ Viewer trong Crystal Reports cho phép xem trước báo biểu và sau đó nhấn vào nút Print để in báo biểu, nhưng sẽ có lúc bạn cần in trực tiếp mà không phải hiển thị Viewer. Sử dụng mô hình đối tượng *ReportDocument*, bạn có thể in trực tiếp kết xuất của báo biểu ra máy in. Bạn còn có thể điều khiển chọn máy in, số trang in, v.v... Trong phần tiếp theo, chúng ta sẽ tìm hiểu kỹ thuật in bằng mô hình đối tượng *ReportDocument*.

Phương pháp dễ nhất để in báo biểu là gọi phương thức *PrintToPrinter*. Phương thức này có thể được sử dụng để in báo biểu ra một máy in mặc định. Phương thức này sử dụng thêm các thuộc tính sau:

- ♦ Number of Copies (số bản copy).
- ♦ Start Page (trang bắt đầu).
- ♦ End Page (trang kết thúc).

Ví dụ dưới đây sẽ sử dụng phương thức này để in báo biểu ra máy in mặc định:

1. Mở Visual Studio, click chọn File | New | Project.
2. Từ cửa sổ New Project, click vào mục Visual Studio Project và chọn Windows Application.
3. Nhập tên cho dự án và click OK. Trong ví dụ này chúng ta đặt tên dự án là VB_PrintReport. Một dự án mới sẽ xuất hiện cùng cửa sổ Form trống.

- Đầu tiên cần chỉ rõ báo biểu chúng ta muốn làm việc. Để thêm một báo biểu vào dự án, chọn Project | Add Existing Item và chọn file báo biểu có sẵn. Trong ví dụ này ta chọn báo biểu mẫu InvoiceReport.rpt trong thư mục Projects trên đĩa CD ROM.
- Viết mã cho sự kiện Form Load như sau:

```
Dim InvoiceReport as New InvoiceReport()
```

- Từ cửa sổ ToolBox, kéo một nút Button vào Form.
- Viết mã cho sự kiện Click của nút nhấn như sau:

```
InvoiceReport.PrintToPrinter(1, True, 0, 0)
```

- Cuối cùng, click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Báo biểu bây giờ sẽ được in trực tiếp ra máy in mặc định mà không cần hiển thị Viewer.
- Nếu muốn in ra một máy in khác (trường hợp máy tính kết nối với nhiều máy in) bạn có thể chỉ định thêm thông số máy in như sau:

```
With InvoiceReport.PrintOptions
```

```
.PrinterName= "Canon Pixma iP1000"
```

```
.PaperSize=CrystalDecisions.
```

```
[Shared].PaperSize.PaperA4
```

```
.PaperOrientation = CrystalDecisions.[Shared].
```

```
PaperOrientation.Landscape
```

```
End with
```

Đoạn mã trên yêu cầu chương trình in báo biểu ra máy in Canon trên khổ giấy A4 theo hướng in ngang.

3.3. Xuất dữ liệu báo biểu

Bạn có thể kết xuất nội dung báo biểu ra file theo các định dạng khác. Quá trình này gọi là xuất dữ liệu (export).

Có hai phương thức kết xuất khác nhau - thứ nhất bạn gọi phương thức *ExportToDisk* và chỉ rõ dạng file muốn kết xuất. Thứ hai, là phương thức *Export* sử dụng phức tạp hơn với các tùy chọn kết xuất mở rộng.

Đối tượng *ReportDocument* hỗ trợ những dạng kết xuất sau:

Định dạng	Dạng kết xuất
<i>Adobe Acrobat PDF</i>	<i>PortableDocFormat</i>
<i>Crystal Reports</i>	<i>CrystalReport</i>
<i>HTML 3.2</i>	<i>HTML32</i>
<i>HTML4.0</i>	<i>HTML40</i>
<i>Microsoft Excel</i>	<i>ExcelRecord</i>
<i>Microsoft Word</i>	<i>WordForWindows</i>
<i>No Format</i>	<i>NoFormat</i>
<i>Rich Text</i>	<i>RichText</i>
<i>Rich Text (Editable)</i>	<i>EditableRTF</i>
<i>Text</i>	<i>Text</i>
<i>Text(CharacterSeparated)</i>	<i>CharacterSeparatedValues</i>
<i>Text (Tab Separated)</i>	<i>TabSeparatedText</i>

Để sử dụng phương thức *ExportToDisk*, tất cả những gì bạn cần làm là chỉ định dạng kết xuất và đường dẫn file cần lưu. Để sử dụng phương thức *ExportToDisk*, bạn thực hiện các bước sau:

1. Mở Visual Studio, click chọn File | New | Project.
2. Từ cửa sổ New Project, click vào mục Visual Studio Project và chọn Windows Application.
3. Nhập tên cho dự án và click OK. Trong ví dụ này chúng ta đặt tên dự án là VB_SimpleExportReport. Một dự án mới sẽ xuất hiện cùng cửa sổ Form trống.
4. Đầu tiên bạn cần chỉ định báo biểu chúng ta muốn làm việc. Để thêm một báo biểu vào trong dự án, chọn Project | Add Existing Item và trở đến file báo biểu cần sử dụng. Trong ví dụ này chúng ta sử dụng báo biểu mẫu *ExportReport.rpt* chứa trong thư mục Projects trên đĩa CD ROM.
5. Từ cửa sổ công cụ Toolbox, kéo đối tượng *ReportDocument* vào Form – Visual Studio sẽ mở một hộp thoại với danh sách các báo biểu đang có trong dự án. Hãy chọn báo biểu chúng ta vừa thêm vào.

6. Bạn viết mã cho sự kiện Form Load như sau:

```
Dim ReportToExport as New exportReport()  
ReportToExport.ExportToDisk  
(CrystalDescisions.[Shared].ExportFormatType.Excel,  
"c:\simpleexport.xls")
```

7. Click Debug / Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Báo biểu bây giờ sẽ được kết xuất ra file simpleexport.xls theo định dạng Excel.
8. Phương thức ExportToDisk sử dụng nhanh và khá dễ, nhưng nếu bạn muốn điều khiển quá trình xử lý kết xuất chi tiết hơn thì phải sử dụng phương thức *Export*. Phương thức *Export* cho phép bạn chọn thiết bị mà file sẽ kết xuất.

Nơi lưu trữ	Vị trí
<i>Disk</i>	<i>DiskFile</i>
<i>Exchange Folder</i>	<i>ExchangeFolder</i>
<i>Email (MAPI)</i>	<i>MicrosoftMail</i>
<i>No Destination</i>	<i>NoDestination</i>

Dưới đây là ví dụ kết xuất báo biểu sang dạng PDF và lưu nó vào ổ đĩa cứng bằng phương thức *Export*:

1. Mở Visual Studio, click chọn File | New | Project.
2. Từ cửa sổ New Project, click vào mục Visual Studio Project và chọn Windows Application.
3. Nhập tên cho dự án và click OK. Trong ví dụ này chúng ta đặt tên dự án là VB_ExportReport. Một dự án mới sẽ xuất hiện cùng cửa sổ Form trống.
4. Đầu tiên bạn cần chỉ định báo biểu chúng ta muốn làm việc. Để thêm một báo biểu vào trong dự án, chọn Project | Add Existing Item và trỏ đến file báo biểu cần sử dụng. Trong ví dụ này chúng ta sử dụng báo biểu mẫu ExportReport.rpt chứa trong thư mục Projects trên đĩa CD ROM.
5. Từ cửa sổ công cụ ToolBox, kéo đối tượng ReportDocument vào Form – Visual Studio sẽ mở một hộp thoại với danh sách các báo biểu đang có trong dự án. Hãy chọn báo biểu chúng ta vừa thêm vào.

6. Bạn viết mã cho sự kiện Form Load như sau:

```
Dim ReportToExport As New ReportToExport
    ReportToExport.Load()
Dim ExportOptions As New
    CrystalDescisions.Shared.ExportOptions
Dim DiskFileDestinationOptions As New
    CrystalDescisions.Shared.

    DiskFileDestinationOptions
Dim FormatTypeOptions As New
    CrystalDescisions.Shared.

    PdfRtfWordFormatOptions
ExportOptions = ReportToExport.ExportOptions

With FormatTypeOptions
    .FirstpageNumber=1
    .LastPageNumber=10
End with

With DiskFileDestinationOptions
    .DiskFileName="c:\advancedexport.pdf"
End with

With ExportOptions
    .ExportDestinationType=CrystalDescisions.

    Shared.ExportFormatType.PortableDocFormat
    .ExportDestinationOptions=

    DiskFileDestinationOptions
    .FormatOptions=FormatTypeOptions
End with

ReportToExport.Export()
```

7. Sau cùng, click Debug! Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Báo biểu sẽ được kết xuất và lưu vào ổ đĩa cứng C. Theo cách tương tự bạn có thể kết xuất chuyển báo biểu sang Mail hay thư mục Mail Server của Microsoft Exchange.

3.4. Làm việc với các tham số

Mô hình *ReportDocument* cung cấp khả năng làm việc với các Field tham số trong báo biểu tương tự như với mô hình *Report Viewer*.

Trong ví dụ sau tiếp theo, chúng ta sẽ học cách đọc các tham số trong báo biểu bằng mô hình ReportDocument. Chương trình sẽ hiển thị các giá trị mặc định và chuỗi mô tả liên quan đến các Field tham số có trong báo biểu. Người sử dụng có thể chọn giá trị cho các Field tham số và chương trình sẽ chuyển các giá trị này ngược lại cho báo biểu.

1. Mở Visual Studio, click chọn File | New | Project.
2. Từ cửa sổ New Project, click vào mục Visual Studio Project và chọn Windows Application.
3. Nhập tên cho dự án và click OK. Trong ví dụ này chúng ta đặt tên dự án là VB_ParameterValues. Một dự án mới sẽ xuất hiện cùng cửa sổ Form trống.
4. Đầu tiên bạn cần chỉ định báo biểu chúng ta muốn làm việc. Để thêm một báo biểu vào trong dự án, chọn Project | Add Existing Item và trở đến file báo biểu cần sử dụng. Trong ví dụ này chúng ta sử dụng báo biểu mẫu Citysales.rpt chứa trong thư mục Projects trên đĩa CD ROM.
5. Từ cửa sổ công cụ ToolBox, kéo đối tượng ReportDocument vào Form – Visual Studio sẽ mở một hộp thoại với danh sách các báo biểu đang có trong dự án. Hãy chọn báo biểu chúng ta vừa thêm vào.
6. Tiếp theo, thêm 5 đối tượng nhân Label, ComboBox lên Form để hiển thị các giá trị của tham số trong báo biểu.
7. Thêm tiếp vào Form nút Button đặt thuộc tính Text của nút là View Report.
8. Cuối cùng, thêm vào Form điều khiển Crystal Reports Viewer. Bây giờ bạn có thể bắt đầu viết mã cho Form.
9. Double click vào Form và khai báo các biến trong lớp Form như sau:

```
Dim ParameterFieldDefinitions As
    CrystalDecisions.CrystalReports.Engine.
        ParameterFieldDefinitions
Dim ParameterFieldDefinition As
    CrystalDecisions.CrystalReports.Engine.
        ParameterFieldDefinition
Dim defaultParameterValues As
    CrystalDecisions.Shared.ParameterValues
Dim defaultParameterValue As
    CrystalDecisions.Shared.ParameterValue
```

```

Dim currentParameterValues As
    CrystalDecisions.Shared.ParameterValues
Dim currentParameterValue As
    CrystalDecisions.Shared.ParameterValue
Dim ParameterDiscreteValue As
    CrystalDecisions.Shared.ParameterDiscreteValue
Dim ParameterDiscreteValues As
    CrystalDecisions.Shared.ParameterDiscreteValue
Dim ReportDocumentwithParameters As
    CrystalDecisions.CrystalReports.Engine.
        ReportDocument

```

10. Mã của sự kiện Form Load sẽ được cài đặt như sau để duyệt qua danh sách các tham số có sẵn trong báo biểu:

```

ParameterFieldDefinitions =
    ReportDocumentwithParameters.
        DataDefinition.ParameterFields
For i As Integer= 0 To
    ParameterFieldDefinitions.Count - 1
    ParameterFieldDefinition =
        ParameterFieldDefinitions(i)
    defaultParameterValues =
        ParameterFieldDefinition.DefaultValues
    If i = 0 Then
        Label1.Text =
            ParameterFieldDefinition.PromptText.ToString
        For Each defaultParameterValue In
            defaultParameterValues
            ParameterDiscreteValues =
                CType(defaultParameterValue,
                    CrystalDecisions.Shared.ParameterValue)
            ComboBox1.Items.Add(
                ParameterDiscreteValues.Value.ToString())
        Next
    End If
    If i = 1 Then
        Label2.Text = ParameterFieldDefinition.
            PromptText.ToString
        For Each defaultParameterValue In
            defaultParameterValues
            ParameterDiscreteValue =
                CType(defaultParameterValue,
                    CrystalDecisions.Shared.ParameterDiscreteValue)
            ComboBox2.Items.Add(
                ParameterDiscreteValue.Value.ToString())
        Next
    End If
Next

```

```

End If
If i = 2 Then
    Label3.Text =
ParameterFieldDefinition.PromptText.ToString
    For Each defaultParameterValue In
        defaultParameterValues
        ParameterDiscreteValue =
            CType(defaultParameterValue,
                CrystalDecisions.Shared.
                    ParameterDiscreteValue)
        ComboBox3.Items.Add(
            ParameterDiscreteValue.Value.ToString())
    Next
End If
If i = 3 Then
    Label4.Text = ParameterFieldDefinition.
        PromptText.ToString
    For Each defaultParameterValue In
        defaultParameterValues
        ParameterDiscreteValue =
            CType(defaultParameterValue,
                CrystalDecisions.Shared.
                    ParameterDiscreteValue)
        ComboBox4.Items.Add(
            ParameterDiscreteValue.Value.ToString())
    Next
End If
If i = 4 Then
    Label5.Text = ParameterFieldDefinition.
        PromptText.ToString
    For Each defaultParameterValue In
        defaultParameterValues
        ParameterDiscreteValue =
            CType(defaultParameterValue,
                CrystalDecisions.Shared.
                    ParameterDiscreteValue)
        ComboBox5.Items.Add(
            ParameterDiscreteValue.Value.ToString())
    Next
End If
Next

```

11. Tiếp theo, chúng ta viết mã cho sự kiện Click của nút nhấn để chuyển giá trị mà người dùng chọn cho tham số trong báo biểu:

```

ParameterFieldDefinitions =
ReportDocument.DataDefinition.ParameterFields

```

```
Dim i As Integer
For i = 0 To ParameterFieldDefinitions.Count - 1
    ParameterFieldDefinition =
        ParameterFieldDefinitions(i)
    currentParameterValues =
        ParameterFieldDefinition.CurrentValues
    Dim ParameterDiscreteValue As
        CrystalDecisions.Shared.ParameterDiscreteValue
        = New CrystalDecisions.Shared.
            ParameterDiscreteValue
    If i = 0 Then
        ParameterDiscreteValue.Value =
            ComboBox1.SelectedItem
        currentParameterValues.Clear()
        currentParameterValues.Add(
            ParameterDiscreteValue)
        ParameterFieldDefinition.
            ApplyCurrentValues(currentParameterValues)
    End If
    If i = 1 Then
        ParameterDiscreteValue.Value =
            ComboBox2.SelectedItem
        currentParameterValues.Clear()
        currentParameterValues.Add(
            ParameterDiscreteValue)
        ParameterFieldDefinition.
            ApplyCurrentValues(currentParameterValues)
    End If
    If i = 2 Then
        ParameterDiscreteValue.Value =
            ComboBox3.SelectedItem
        currentParameterValues.Clear()
        currentParameterValues.Add(
            ParameterDiscreteValue)
        ParameterFieldDefinition.
            ApplyCurrentValues(currentParameterValues)
    End If
    If i = 3 Then
        ParameterDiscreteValue.Value =
            ComboBox4.SelectedItem
        currentParameterValues.Clear()
        currentParameterValues.Add(
            ParameterDiscreteValue)
        ParameterFieldDefinition.
            ApplyCurrentValues(currentParameterValues)
    End If
```

```
If i = 4 Then
    ParameterDiscreteValue.Value =
        ComboBox5.SelectedItem
    currentParameterValues.Clear()
    currentParameterValues.Add(
        ParameterDiscreteValue)
    ParameterFieldDefinition.
        ApplyCurrentValues(currentParameterValues)
End If
Next
CrystalReportViewer1.ReportSource = ReportDocument
```

12. Mọi thứ đã sẵn sàng, click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Các nhãn sẽ hiển thị chuỗi mô tả tham số còn các đối tượng ComboBox sẽ hiển thị giá trị tham số. Chọn giá trị cho tham số và click vào nút View Report, báo biểu sẽ hiển thị trong Viewer với nội dung lọc theo giá trị do bạn chọn.

3.5. Đăng nhập tự động vào cơ sở dữ liệu

Trong phần tìm hiểu về mô hình ReportViewer, chúng ta đã đề cập đến cơ chế đăng nhập tự động vào nguồn cơ sở dữ liệu. Mô hình ReportViewer không cho phép tự động đăng nhập một lần. Bạn phải chỉ định tường minh tên bảng cùng với thông tin đăng nhập trong lớp LogonInfo. Điều này có nghĩa là nếu bạn có nhiều bảng liên kết trong cùng một báo biểu thì bạn phải chỉ định thông tin đăng nhập nhiều lần. Mô hình đối tượng ReportDocument cho phép chúng ta duyệt qua tất cả các nguồn dữ liệu khác nhau đang sử dụng trong báo biểu và xác định thông số đăng nhập chỉ một lần.

Trong dự án mẫu tiếp theo, chúng ta sẽ duyệt qua tất cả các bảng tồn tại bên trong một báo biểu và thiết lập thông tin kết nối cho chúng.

1. Mở Visual Studio, click chọn File | New | Project.
2. Từ cửa sổ New Project, click vào mục Visual Studio Project và chọn Windows Application.
3. Nhập tên cho dự án và click OK. Trong ví dụ này chúng ta đặt tên dự án là VB_SecureDatasource. Một dự án mới sẽ xuất hiện cùng cửa sổ Form trống.
4. Đầu tiên bạn cần chỉ định báo biểu chúng ta muốn làm việc. Để thêm một báo biểu vào trong dự án, chọn Project | Add Existing Item và trở đến file báo biểu cần sử dụng. Trong ví dụ này chúng ta sử dụng

báo biểu mẫu SecureReport.rpt chứa trong thư mục Projects trên đĩa CD ROM.

5. Từ cửa sổ công cụ ToolBox, kéo đối tượng ReportDocument vào Form – Visual Studio sẽ mở một hộp thoại với danh sách các báo biểu đang có trong dự án. Hãy chọn báo biểu chúng ta vừa thêm vào.
6. Cũng từ cửa sổ công cụ ToolBox, kéo đối tượng CrystalReportViewer vào trong Form và đặt thuộc tính Docking là Full Screen.
7. Tiếp theo, double click vào Form để viết mã cho sự kiện Form Load của Form liên kết đến cơ sở dữ liệu Northwind chuẩn của SQL Server như sau:

```
Dim SecureReport As SecureReport
```

```
Dim tables As  
    CrystalDecisions.CrystalReports.Engine.Table
```

```
Dim table As  
    CrystalDecisions.CrystalReports.Engine.Table
```

```
Dim tableLogonInfo As  
    CrystalDecisions.Shared.TableLogOnInfo
```

```
Dim LogonInfos As  
    CrystalDecisions.Shared.TableLogOnInfo
```

```
Dim LogonInfo As  
    CrystalDecisions.Shared.TableLogOnInfo
```

```
Dim ConnectionInfo As  
    CrystalDecisions.Shared.ConnectionInfo =  
New  
    CrystalDecisions.Shared.ConnectionInfo
```

```
ConnectionInfo.ServerName="(local)"  
ConnectionInfo.DatabaseName="Northwind"  
ConnectionInfo.UserID="Sa"  
ConnectionInfo.Password="Admin"  
table =SecureReport.Database.Table
```

```
For each table in tables  
    tableLogonInfo=table.LogOnInfo  
    tableLogonInfo.ConnectionInfo=ConnectionInfo  
    table.ApplyLogonInfo(tableLogonInfo)
```

Next

```
CrystalReportViewer1.ReportSource=SecureReport
```

8. Sau khi hoàn thành, click Debug | Start hoặc nhấn phím F5 để bắt đầu thực thi ứng dụng. Cơ sở dữ liệu sẽ được kết nối, thông tin đăng nhập sẽ được gán cho mỗi bảng. Báo biểu sẽ thực thi mà không bị lỗi đang nhập như trước đó.

4. TÓM TẮT

Chương này đã hướng dẫn bạn cách tích hợp báo biểu vào các ứng dụng .NET của môi trường lập trình ứng dụng Visual Studio. Bạn đã biết cách sử dụng hai mô hình ReportViewer và ReportDocument cùng với kỹ thuật chuyển tham số bằng mã lệnh chương trình cho hai mô hình này. Chương tiếp theo chúng ta sẽ học các tích hợp báo biểu Crystal Reports vào một loại ứng dụng khá phổ biến trong thế giới Internet ngày nay, đó là ứng dụng Web.

Chương 10:

TÍCH HỢP BÁO BIỂU VÀO ỨNG DỤNG WEB

- ❖ *Tổng quan.*
- ❖ *Tiếp cận ứng dụng Web.*
- ❖ *Làm việc với Report View Object Model.*
- ❖ *Tóm tắt.*

1. TỔNG QUAN

Khi Internet trở nên thông dụng, mọi người nhận ra ngay đây là một công cụ mạnh kết nối con người với thông tin; các nhà phát triển ứng dụng lại nhìn thấy một khả năng dùng trình duyệt Web (Web Browse) để chạy ứng dụng ở mọi nơi mà không cần cài đặt bất cứ thứ gì lên máy cục bộ.

Khi giới thiệu công nghệ Active Server Pages (ASP), Microsoft đã tạo nên một công cụ thiết kế Web cho các nhà phát triển, giúp họ có thể xây dựng các trang Web động thay cho mã HTML tĩnh. Năm 2002, Microsoft xây dựng phát triển nền .NET Framework cùng phiên bản ASP.NET (bản năng cấp của ASP). Phiên bản này chạy trong môi trường CLR (Common Language Runtime) của .NET và hỗ trợ rất nhiều ngôn ngữ lập trình. Là một đối tác thân thiết của Microsoft, Crystal Reports đã được tích hợp vào Visual Studio .NET từ phiên bản 2002 trở đi, giờ đây báo biểu có thể được xem và tương tác từ ứng dụng Web ASP.NET. Chương này sẽ hướng dẫn bạn cách tích hợp Crystal Reports vào ứng dụng ASP.NET được tạo và sử dụng bởi Visual Studio 2005 và .NET Framework.

Tất cả các ví dụ ứng dụng và báo biểu mẫu của chương này đều nằm trên đĩa CD ROM kèm theo giáo trình.

2. TIẾP CẬN ỨNG DỤNG WEB

Giống như với các ứng dụng Windows, đã được trình bày trong chương 9, bạn có thể dùng hai mô hình đối tượng sau đây để tích hợp báo biểu vào trong ứng dụng Web ASP.NET của mình. Mô hình đối tượng thứ

nhất là CrystalReportViewer, đối tượng này sẽ cung cấp tất cả các thuộc tính, phương thức và sự kiện mà bạn cần khi xem báo biểu. ReportViewer trên Web có thể gần giống với ReportViewer của các ứng dụng Windows; tuy một số thuộc tính hay phương thức của đối tượng có thể khác nhau, nhưng về mặt ý nghĩa thì chúng hoàn toàn giống nhau và thực hiện cùng một mục đích: cho phép xem báo biểu từ các ứng dụng của bạn và cho phép thay đổi các kết xuất báo biểu. Để có thể xử lý báo biểu về mọi mặt và toàn quyền điều khiển báo biểu, bạn cần sử dụng mô hình đối tượng thứ hai là ReportDocument.

Ghi chú: Mô hình đối tượng ReportDocument trên Web này giống với mô hình đối tượng ReportDocument mà chúng ta đã học trong chương trước. Bạn sẽ thấy một số nét giống nhau giữa chúng ở cuối chương nhưng đa phần là rất nhiều thuộc tính khác nhau. Hầu hết sự khác nhau nếu có là do ứng dụng Windows và ứng dụng Web có cách cư xử khác biệt và đó là lý do Crystal sử dụng hai mô hình đối tượng ReportDocument cho cùng một mục đích.

Sử dụng mô hình đối tượng nào là tùy vào kiểu ứng dụng Web mà bạn phát triển. Trong chương 8, chúng ta đã xem qua các kiểu tích hợp mà bạn có thể sử dụng trong ứng dụng của mình như: xem và hiển thị nội dung báo biểu, truyền biến tham số cho báo biểu. Đối với các ứng dụng Web đơn giản, mô hình đối tượng CrystalReportView là sự lựa chọn tốt nhất, do nó cung cấp tất cả các lựa chọn cơ bản cần thiết để bạn có thể xem (Preview), thiết lập các mẫu tin sẽ được lọc bên trong ứng dụng của mình. Bạn có thể xem hình dạng của báo biểu trên trang Web Form như hình 10.1.

Order ID	Customer Name	City	Region	Order Date	Order Amount
1000000001	7 Days Far 7 Brothers	Los Angeles	CA	10/25/2001	\$100.00
1000000002	Against The Wind Blues	Rome	NY	10/25/2001	\$479.00
1000000003	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000004	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000005	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000006	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000007	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000008	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000009	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000010	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000011	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000012	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000013	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000014	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000015	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000016	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000017	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000018	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000019	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000020	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000021	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000022	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000023	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000024	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000025	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000026	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000027	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000028	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000029	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000030	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000031	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000032	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000033	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000034	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000035	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000036	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000037	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000038	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000039	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000040	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000041	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000042	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000043	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000044	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000045	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000046	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000047	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000048	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000049	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00
1000000050	Against The Wind Blues	Hong Kong GAO	MA	10/25/2001	\$100.00

Hình 10.1: ReportViewer của Crystal Reports trên Web.

Chú ý rằng, đối với những người đã quen sử dụng các ứng dụng Windows, khi xem báo biểu trên Web sẽ có một chút khác biệt nhỏ. Thanh công cụ Viewer phía trên và phía dưới màn hình không thể neo được trên cửa sổ Web Form. Tuy nhiên, Viewer này có thể chia sẻ các chức năng không khác gì Windows ReportViewer, kể cả việc truyền biến tham số cho Viewer cũng như đăng ký login vào CSDL theo cách thông thường của ứng dụng Windows.

Nếu có thêm các nhu cầu tích hợp khác, như kết nối dữ liệu thông qua ADO hoặc ADO .NET, in ra máy in hoặc kết xuất ra các định dạng file chuẩn, điều khiển công thức tính của các trường (Field) hay khuôn dạng báo biểu, bạn cần sử dụng đến mô hình đối tượng ReportDocument.

Trong chương này chúng ta sẽ xem xét kiểu đối tượng CrystalReportsViewer thông qua việc chạy và thử các chức năng có sẵn. Phần cuối chương, chúng ta sẽ tìm hiểu cách sử dụng đối tượng ReportDocument.

Bạn không thể dùng lẫn lộn hai đối tượng này, nếu sau khi xem qua đối tượng CrystalReportViewer và nhận thấy chưa đáp ứng được nhu cầu, bạn có thể phải chuyển sang sử dụng đối tượng ReportDocument còn đối tượng CrystalReportViewer chỉ dùng để xem kết quả của báo biểu.

3. LÀM VIỆC VỚI CRYSTALREPORTVIEWER

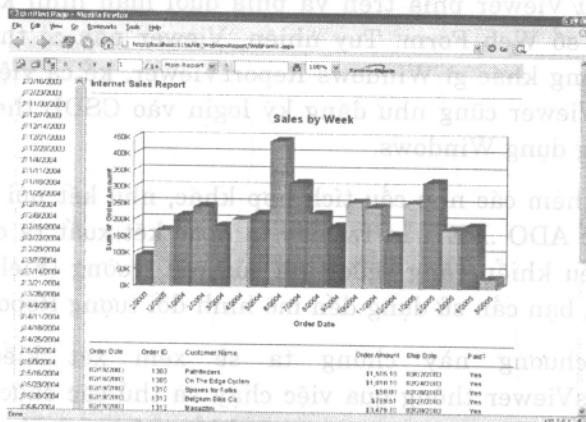
ReportViewer là điều khiển (Control) của .NET. Bạn có thể nắm kéo và thả nó vào trang Web Form, sau đó viết mã thực thi cho các sự kiện của Control. Các đoạn mã này sẽ điều khiển và qui định cách thức thể hiện giao diện của ReportViewer.

3.1. Xem báo biểu trên Web

Để xem báo biểu trong ứng dụng ASP.NET, bạn chỉ cần viết một dòng mã. Kéo CrystalReportViewer vào Web Form của bạn và thiết lập thuộc tính ReportSource trỏ đến file báo biểu (nếu đã thiết lập thuộc tính này trong cửa sổ Properties thì bạn cũng chẳng phải viết thêm dòng mã nào nữa cả).

Sau khi thêm CrystalReportViewer vào Web Form và thiết lập thuộc tính ReportSource, bạn biên dịch và chạy ứng dụng Web Form của mình trong trình duyệt. ReportViewer sẽ hiển thị nội dung báo biểu ngay bên trong trình duyệt Web, hãy xem chúng ta thực hiện điều này như thế nào:

1. Từ Start menu chọn và mở Visual Studio 2005.
2. Nhấn chọn File | New | New Web Site.



Hình 10.2: Báo biểu hiển thị trên Web.

3. Nhập vào tên dự án và chọn nơi cất giữ dự án, sau đó chọn OK. Ví dụ: dự án này được đặt tên là VB_WebViewerReport.
4. Một dự án Web mới sẽ được tạo ra với một Web Form trống. Bạn chọn View | Toolbox, nắm kéo CrystalReportViewer vào trang Web Form.
5. Double-click vào Form để vào phần viết mã cho trang, bạn thêm mã vào thủ tục Form_Load nạp file báo biểu như sau:
CrystalReportViewer1.ReportSource
= "d:\Projects\websales.rpt"
6. Cuối cùng, chọn Debug > Start hoặc nhấn phím F5 để bắt đầu chạy ứng dụng. Cửa sổ trình duyệt sẽ được mở ra để thực thi Web Form và Viewer mà bạn thêm vào (hình 10.2).

Theo mặc định, Viewer hiển thị thanh công cụ ở phía trên trình Preview. Thanh công cụ này bao gồm một số nút như sau:

- ♦ Export: Dùng để xuất báo biểu ra các định dạng World, Excel, PDF thông qua một hộp thoại xuất.
- ♦ Print: Dùng mở hộp thoại in.
- ♦ Group Tree: Hiển thị hoặc ẩn nhóm cây thư mục bên cạnh trái của bản báo biểu.

- ♦ Page Navigation Button: Cung cấp nút điều hướng cho các trang báo biểu.
- ♦ Find: Dùng mở hộp thoại tìm kiếm cho phép nhập chuỗi để tìm trong báo biểu.
- ♦ Zoom: Chức năng phóng đại.

Ngoài ra bạn có thể tùy biến cách hiển thị của Viewer như chúng ta sẽ xem xét trong phần sau.

3.2. Thay đổi cách thức hiển thị của Viewer

Viewer cung cấp rất nhiều thuộc tính để bạn thay đổi diện mạo của nó. Nhóm thuộc tính đầu tiên xử lý các diện mạo bên ngoài của Viewer; còn nhóm thuộc tính thứ hai sẽ xử lý các biểu tượng Icon xuất hiện trên thanh công cụ Viewer. Các thiết lập ban đầu của thuộc tính được thể hiện trong bảng sau:

Phương thức	Mô tả	Kiểu
DisplayToolbar	Các Control hiển thị phía trên thanh công cụ của Viewer.	Boolean
DisplayBottomToolbar	Các Control hiển thị phía dưới thanh công cụ của Viewer.	Boolean
DisplayGroupTree	Control hiển thị Group Tree trên cạnh trái của Viewer.	Boolean
PageToTreeRation	Cách mà Control Group Tree sẽ xuất hiện trong trang báo biểu.	Integer

Nếu bạn không muốn hiển thị thanh công cụ của Viewer và điều chỉnh cây phân nhóm (Group Tree) nhỏ hơn, bạn có thể thêm đoạn mã sau vào trong Form_Load:

```
CrystalReportViewer1.DisplayToolbar = False  
CrystalReportViewer1.DisplayBottomToolbar = False  
CrystalReportViewer1.PageToTreeRadio = 8
```

Nếu bạn sử dụng đoạn mã trên để tắt thanh công cụ và thay đổi tỉ lệ Ration, Viewer sẽ hiển thị báo biểu như hình 10.3.



Hình 10.3: Viewer báo biểu đã loại bỏ thanh công cụ với kích thước Group Tree tách rời.

Bằng cách điều chỉnh các biểu tượng trên thanh công cụ, bạn có thể điều chỉnh cách thức tương tác của người sử dụng với báo biểu. Ví dụ: Nếu bạn không muốn cho hiển thị nút in, người sử dụng sẽ không thể nào in báo biểu từ Viewer. Bạn có thể thay đổi các thuộc tính sau đây để làm xuất hiện hay giấu đi các nút của thanh công cụ:

- ◆ HasCrystalLogo.
- ◆ HasDrillUpButton.
- ◆ HasExportButton.
- ◆ HasPageNavigationButtons.
- ◆ HasPrintButton.
- ◆ HasRefreshButton.

- ◆ HasSearchButton.
- ◆ HasToggleGroupTreeButton.
- ◆ HasViewList.
- ◆ HasZoomFactorList.

Tất cả những thuộc tính trên đều có kiểu **Boolean**, để có thể ẩn đi nút in ấn, bạn có thể viết mã như sau:

```
CrystalReportViewer1.HasPrintButton = False
```

Khi xem báo biểu, Viewer sẽ ẩn nút in và hiển thị như hình 10.4 sau đây.



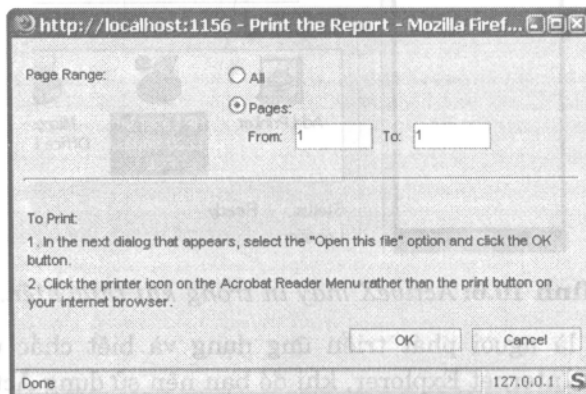
Hình 10.4: Viewer báo biểu đã loại bỏ nút in ấn.

Ngoài các thuộc tính thay đổi các nút nhấn trên thanh công cụ, bạn còn có thể sử dụng các thuộc tính để định dạng nội dung của khung Group Tree xuất hiện bên cạnh trái của cửa sổ Viewer. Group Tree giúp người sử dụng di chuyển qua các mục của báo biểu được dễ dàng hơn, nhưng cũng sẽ có đôi lúc bạn muốn thay đổi cách thể hiện của nó. Để làm điều này bạn có thể sử dụng thuộc tính GroupTreeStyle. Chẳng hạn như để thay đổi font hiển thị của Group Tree bạn viết lệnh sau:

```
CrystalReportViewer1.GroupTreeStyle.  
Font.Name = "Arial"
```

3.3. Chọn lựa chế độ in ấn

Một trong những tính năng mới trong Crystal Reports XI là có khả năng in trực tiếp báo biểu từ trang Web bằng Viewer. Tính năng in này có thể thực hiện theo 2 cách. Cách thứ nhất là sử dụng file PDF (Adobe® Acrobat®) như một file trung gian, với chức năng in ấn được cung cấp thông qua Adobe Acrobat Reader. Kỹ thuật này yêu cầu Adobe Acrobat Reader đã được cài đặt trên máy tính của người sử dụng từ trước. Khi người sử dụng nhấp vào nút in ấn, một hộp thoại sẽ hiển thị như hình 10.5 hướng dẫn bạn cách in báo biểu ra Acrobat Reader.



Hình 10.5: Hộp thoại in báo biểu.

Do hầu hết người sử dụng đều đã quen với file PDF, đây là phương thức in ấn mặc định và không đòi hỏi phải thêm bất kỳ phần mềm nào vào hoặc cài đặt các thành phần (component) khác.

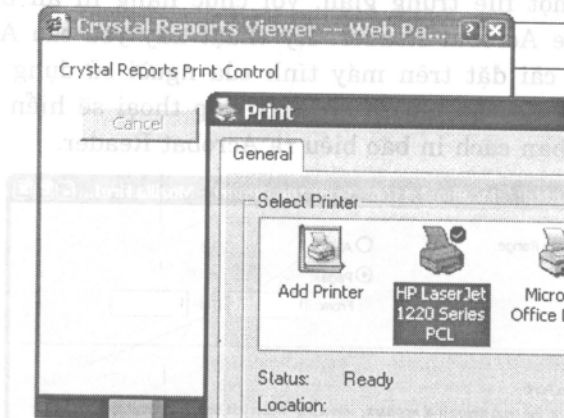
Phương thức in ấn thứ hai là thông qua ActiveX control từ Internet Explorer, thành phần ActiveX này sẽ in báo biểu trực tiếp ra máy in của bạn. Khi thành phần này được kích hoạt và người sử dụng nhấp vào nút in trên Viewer của báo biểu, ActiveX control tự động tải về (download) và tự cài đặt vào môi trường trình duyệt. Một cửa sổ trung gian nhỏ giống như hình 10.6 hiển thị và hộp thoại của sổ máy in chuẩn được mở ra. Trong hộp thoại này người sử dụng có thể chọn lựa và định nghĩa máy in (chỉ xuất dành cho trình duyệt Internet Explorer).

Bạn có thể điều chỉnh cách thức in thông qua thuộc tính **PrintMode** của Viewer. Chúng ta có thể sử dụng 2 chế độ in sau đây:

- ◆ CrystalDecisions.Web.PrintMode.Pdf.
- ◆ CrystalDecisions.Web.PrintMode.ActiveX.

Để thiết lập chế độ in với ActiveX và in trực tiếp ra máy in, bạn viết mã như sau:

```
CrystalReportViewer1.PrintMode =  
CrystalDecisions.Web.PrintMode.ActiveX
```



Hình 10.6: ActiveX máy in trong khi thực hiện.

Nếu bạn là người phát triển ứng dụng và biết chắc ứng dụng của mình chạy trên Internet Explorer, khi đó bạn nên sử dụng ActiveX control.

3.4. Hiệu chỉnh việc lọc các mẫu tin báo biểu

Record Selection là quá trình xử lý của Crystal Reports dùng để lọc các mẫu tin báo biểu. Công việc này được thực hiện thông qua đoạn mã sau đây:

`{Customer.Country} = "USA"`

Công thức Record-Selection được viết bởi cú pháp Crystal và có thể tạo ra trong quá trình thiết kế báo biểu hoặc ngay thời điểm xem báo biểu. Bạn có thể xem công thức Record Selection hiện hành đang sử dụng cho báo biểu bằng cách mở báo biểu trong Crystal Reports và chọn **Report > Selection Formulas > Record** để mở trình biên soạn công thức lọc Record Selection.

Bất kỳ công thức lọc Record Selection nào mà bạn xây dựng trong quá trình biên soạn này cũng được lưu vào trong báo biểu. Khi tích hợp các báo biểu vào ứng dụng, bạn có thể sử dụng hai thuộc tính để thiết lập Record Selection khi báo biểu đang chạy. Thuộc tính đầu tiên là SelectionFormula (công thức chọn lọc), có thể được sử dụng để lấy hoặc thiết lập công thức lọc mẫu tin. Thuộc tính thứ hai là ViewTimeSelectionFormula, được sử dụng để gắn thêm công thức lọc vào công thức lọc Record Selection đã tồn tại trong báo biểu.

Ví dụ: Nếu báo biểu đã tồn tại công thức Record Selection sau:

`{Customer.Country} = "USA"`

và bạn đã sử dụng thuộc tính ViewTimeSelectionFormula để thiết lập công thức Record Selection để lọc thêm một thành phố khác như sau:

`{Customer.Country} = "NSW"`

kết quả công thức Record Selection sẽ trở thành:

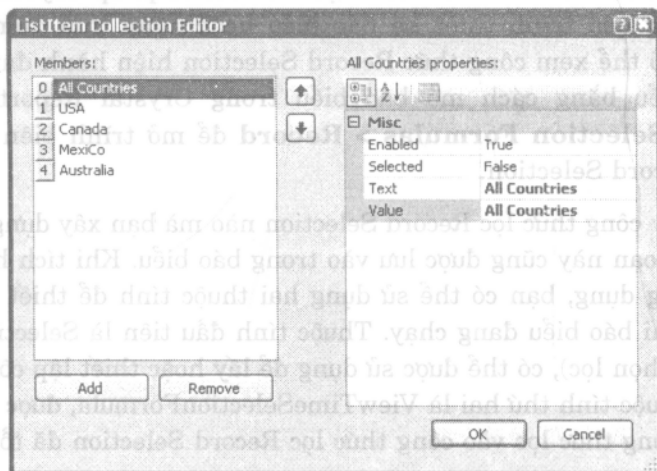
`{Customer.Country} = "USA"`

`And {Customer.Country} = "NSW"`

Chúng ta hãy xem cách ứng dụng ViewTimeSelectionFormula như sau:

1. Từ Start Menu mở Visual Studio 2005 hay phiên bản cao hơn.
2. Chọn **File > New > Web Site**.
3. Nhập tên và chọn vị trí để lưu dự án (Project) của bạn vào và nhấn OK. Trong ví dụ này, chúng ta đặt tên cho dự án là VB_WebRecordSelection.
4. Một dự án mới sẽ được tạo cùng với trang Web form trống có tên là WebForm1.aspx. Chọn **View > Toolbox** và đặt vào Web Form đối tượng ComboBox cùng với nút nhấn Button.

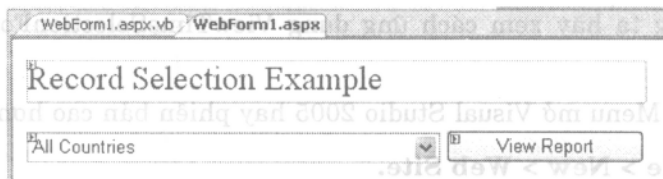
5. Bước kế tiếp, hiệu chỉnh thuộc tính Items của ComboBox như hình 10.7. Hãy nhập các mục sau vào hộp thoại:



Hình 10.7: Trình soạn thảo danh sách.

- ♦ All Countries.
- ♦ USA.
- ♦ Australia.
- ♦ Canada.
- ♦ Mexico.

6. Thuộc tính Text và Value của mỗi mục chọn là giống nhau.
7. Phần tiếp theo, thay đổi thuộc tính Text của nút Button thành **View Report**. Form của bạn bây giờ sẽ trông giống như hình minh họa 10.8.



Hình 10.8: Hoàn thành Form.

8. Từ Form này chúng ta sẽ cho phép người dùng chọn lựa giá trị từ danh sách sổ xuống ComboBox và sau đó click vào nút View Report để di chuyển qua bước kế tiếp của Form, nơi mà giá trị này sẽ được sử dụng để tạo và thiết lập công thức lọc cho báo biểu. Double-click nút nhấn View Report và nhập vào đoạn mã sau:

```
Server.Transfer("WebForm2.aspx", True)
```

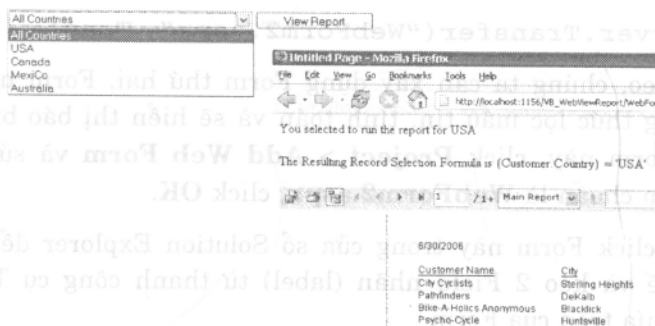
9. Tiếp theo, chúng ta cần xây dựng Form thứ hai, Form này sẽ thiết lập công thức lọc mẫu tin, tính toán và sẽ hiển thị báo biểu. Để xây dựng Form này, click **Project > Add Web Form** và sử dụng luôn tên form chuẩn là **WebForm2.aspx**, click **OK**.
10. Double-click Form này trong cửa sổ Solution Explorer để mở cửa sổ thiết kế và kéo 2 Field nhãn (label) từ thanh công cụ Toolbar vào phần phía trên của Form.
11. Tìm điều khiển CrystalReportViewer và kéo nó lên phần giữa của Form.
12. Khi đã đặt Viewer vào đúng vị trí, double-click vào Form để viết mã cho sự kiện Form_Load như sau:

```
Dim RecordSelectFormula as String
Dim PassedValue as String
PassedValue = Request.Form("DropDownList1")
RecordSelectFormula = "{Customer.Country} = '" &
                        PassedValue & "'"
Label1.Text = "You selected to run the report for " &
              PassedValue
Label2.Text = "The Resulting Record Selection Formula is " &
              RecordSelectFormula
CrystalReportViewer1.ReportSource =
              "d:\Projects\dynselect.rpt"
CrystalReportViewer1.SelectionFormula =
              RecordSelectFormula
```

13. Hiện giờ tất cả những gì còn lại là chạy ứng dụng và xem việc lọc báo biểu của bạn diễn ra như thế nào. Click **Debug > Start** hoặc nhấn phím **F5** để chạy chương trình ứng dụng của bạn. Cửa sổ trình duyệt sẽ hiển thị và ứng dụng Web của bạn sẽ trông như hình minh họa 10.9.
14. Chọn giá trị từ hộp danh sách xổ xuống và click nút nhấn **View Report** để tiến hành thực thi qua trang thứ hai, hiển thị nội dung báo biểu đã áp dụng công thức Record Selection.

Đây chỉ là một ví dụ đơn giản về cách mà mẫu tin được chọn lọc bằng Record Selection. Bạn cần tạo một công thức lọc mẫu tin phức tạp hơn theo nhu cầu của riêng mình.

Record Selection Example

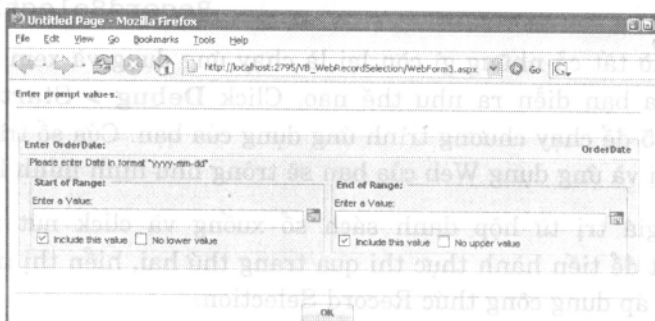


Hình 10.9: Hoàn tất ứng dụng đơn giản.

3.5. Làm việc với Field tham số

Các Field tham số luôn là vấn đề khi vận hành trong các phiên bản Crystal Reports trước đây. Các hộp thoại mặc định yêu cầu nhập giá trị biến rất khó chịu và không dễ sử dụng. Hầu hết các nhà phát triển ứng dụng đều thích tạo lại các hộp thoại nhập giá trị biến tham số hơn là sử dụng hộp thoại mặc định do Crystal cung cấp. Với Crystal Reports XI, những hộp thoại này đã có thêm nhiều cách chọn lựa hơn và có thể sử dụng cả hai loại biến (Parameter) động và phân tầng (Cascading). Bạn có thể đọc giá trị từ một nguồn dữ liệu và sau đó chọn giá trị trả về cho biến tham số trong báo biểu của mình.

Các Field tham số mở rộng cùng hộp thoại nhập tham số bằng giao diện Web (như hình minh họa 10.10) có thể giúp cho các nhà phát triển tạo các báo biểu đơn giản nhanh chóng.



Hình 10.10: Hộp thoại biến tham số mặc định Crystal Reports trên Web.

Nếu muốn kiểm soát hộp thoại tùy biến linh động hơn, hoặc nếu bạn muốn tạo những biến tham số của riêng mình, bạn cần truyền các

giá trị biến tham số cho Viewer và báo biểu thông qua sử dụng lớp `ParameterField`. Lớp này chứa các thuộc tính cho phép đọc và đặt các giá trị thông tin cho biến tham số khi báo biểu thực thi. Ngoài ra, lưu ý Crystal còn cung cấp thêm lớp `ParameterFields` (chú ý có s), là tập hợp lưu giữ tất cả các Field tham số trong báo biểu của bạn.

Để làm việc với các Field tham số, bạn cần phải học cách thức đọc giá trị thông tin từ các Field tham số trong báo biểu của mình.

3.6. Đọc và thiết lập các giá trị rời rạc cho biến tham số

Khi xây dựng các Field tham số trong Crystal Reports, bạn có thể tạo các giá trị chứa trong danh sách tính cho phép người sử dụng chọn lựa khi báo biểu thực thi. Danh sách này cũng có thể chứa giá trị được chọn mặc định mà báo biểu sử dụng.

Để đọc các giá trị biến tham số mặc định của báo biểu, bạn sử dụng thuộc tính `DefaultValues`. Trong ví dụ sau, báo biểu của chúng ta chỉ có một biến tham số, trước hết chúng ta sẽ đọc và gán giá trị cho biến tham số này.

1. Mở Visual Studio 2005.
2. Click **File > New > Web Site**.
3. Nhập vào tên dự án, chọn vị trí lưu dự án của bạn và click **OK**. Trong ví dụ này, chúng ta đặt tên cho dự án là `VB_WebRecordSelection`.
4. Dự án mới sẽ được tạo với một Web Form trống mang tên `WebForm1.aspx`. Click chọn **View > Toolbox** và kéo vào Form một đối tượng nhãn `Label`, cùng một danh sách sổ xuống `Dropdown` và nút nhấn `Button`.
5. Double-click vào Form bạn khai báo biến ngay phía bên dưới định nghĩa lớp **`Inherits System.Web.UI.Page`** như sau:

```
Dim ParameterFields As  
    CrystalDecisions.Shared.ParameterFields  
Dim ParameterField As  
    CrystalDecisions.Shared.ParameterField  
Dim ParameterDiscreteValue As  
    CrystalDecisions.Shared.ParameterDiscreteValue
```

6. Trong thủ tục xử lý sự kiện `Form_Load`, bạn cài đặt cách hiển thị báo biểu và danh sách tham số mặc định như sau:

```
CrystalReportViewer1.ReportSource =
    "D:\projects\DYNPARAM.rpt"
CrystalReportViewer1.DisplayGroupTree = False
ParameterFields =
    CrystalReportViewer1.ParameterFieldInfo
ParameterField = ParameterFields(0)
Label1.Text = ParameterField.PromptText
```

```
Dim i As Integer
For i = 0 To (ParameterField.DefaultValues.Count - 1)
    DropDownList1.Items.Add(
        ParameterField.DefaultValues.Item(i).
            Description)
Next
ParameterDiscreteValue = New
CrystalDecisions.Shared.ParameterDiscreteValue
ParameterDiscreteValue.Value = " "
ParameterField.CurrentValues.Add(
    ParameterDiscreteValue)
```

7. Tiếp đến, chuyển trở về cửa sổ Design của Web Form và đổi tên thuộc tính Text trên nút nhấn thành **View Report**.
8. Double-click vào nút nhấn View Report và nhập vào đoạn mã sau:

```
Server.Transfer("WebForm2.aspx", True)
```

9. Từ cửa sổ Project Explorer, Double-click vào tập in WebForm2.aspx và thêm vào 2 nhãn (label) từ cửa sổ công cụ Toolbox giống như thêm Crystal Reports Viewer control vào vậy.
10. Double-click vào Form mới để viết mã, trước hết bạn khai báo biến ngay bên dưới dòng **Inherits System.Web.UI.Page** như sau:

```
Dim ParameterFields As
    CrystalDecisions.Shared.ParameterFields
Dim ParameterField As
    CrystalDecisions.Shared.ParameterField
Dim ParameterDiscreteValue As
    CrystalDecisions.Shared.ParameterDiscreteValue
```

11. Cài đặt sự kiện Form Load cho Form như sau:

```
CrystalReportViewer1.ReportSource =
    "d:\projects\DYNPARAM.rpt"
Dim PassedValue
```



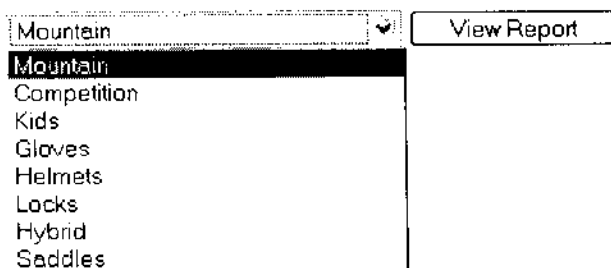
```

PassedValue = Request.Form("DropDownList1")
ParameterFields =
    CrystalReportViewer1.ParameterFieldInfo
ParameterField = ParameterFields(0)
ParameterDiscreteValue = New
    CrystalDecisions.Shared.ParameterDiscreteValue
ParameterDiscreteValue.Value = PassedValue
ParameterField.CurrentValues.Add(ParameterDiscreteValue)

```

12. Click vào **Debug > Start** hoặc nhấn phím **F5** để bắt đầu chạy ứng dụng của bạn. Hộp ComboBox ở phía trên cùng Form sẽ đưa các giá trị mặc định vào trong hộp DropDown của báo biểu như hình minh họa 10.11.

Enter ProductType:



Hình 10.11: Form với giá trị mặc định chứa trong danh sách xổ xuống.

13. Chọn một giá trị bất kỳ và click vào nút View Report. Báo biểu sẽ được lọc lại theo giá trị biến tham số mà bạn vừa chọn

3.6.1. Đặt giá trị phạm vi cho biến tham số

Bạn có thể thiết lập biến tham số cho phép nhập một dãy các giá trị trong khoảng nào đó giống như báo biểu được minh họa trong hình 10.12.

Ad-Hoc Sales Report

Report Parameters:

Order Date: Tuesday, February 18, 2003 to Monday, 2 May, 2005
 Order ID: 1 to 3193
 Salesperson: Aa to Zz

Hình 10.12: Một kiểu báo biểu với biến phạm vi.

Biến tham số này có thể nhận cả hai giá trị: giá trị bắt đầu và giá trị kết thúc.

Đoạn mã sau đây sẽ minh họa cách sử dụng biến tham số truyền theo dạng mảng:

```
CrystalReportViewer1.ReportSource =  
    "d:\projects\RANGEPARAM.rpt"  
ParameterFields =  
    CrystalReportViewer1.ParameterFieldInfo  
ParameterField = ParameterFields(0)  
ParameterRangeValue = New  
    CrystalDecisions.Shared.ParameterRangeValue  
ParameterRangeValue.StartValue = "12/31/2004"  
ParameterRangeValue.EndValue = "01/01/2005"  
  
ParameterField.CurrentValues.Add(ParameterRangeValue)
```

3.7. Cách thức đăng nhập CSDL

Để đăng nhập đến nguồn dữ liệu được đảm bảo hơn từ báo biểu của ứng dụng Web, bạn cần tạo đối tượng LogonInfo và sau đó chỉ rõ tên bảng dữ liệu, tên người sử dụng (user name) và mật khẩu (password). Crystal Reports phiên bản XI này cho phép bạn xử lý đăng nhập bằng mã lệnh chương trình đối với mỗi bảng dữ liệu bạn muốn truy xuất:

```
Dim LogonInfo As CrystalDecisions.Shared.TableLogOnInfo  
CrystalReportViewer1.LogOnInfo = New  
    CrystalDecisions.Shared.TableLogOnInfos  
LogonInfo = New CrystalDecisions.Shared.TableLogOnInfo  
LogonInfo.TableName = "Products"  
LogonInfo.ConnectionInfo.UserID = "Admin"  
LogonInfo.ConnectionInfo.Password = ""  
CrystalReportViewer1.LogOnInfo.Add(LogonInfo)
```

Mô hình đối tượng ReportDocument thường dùng cơ chế này để duyệt qua tất cả các bảng dữ liệu trong báo biểu.

4. TÓM TẮT

ASP.NET là nền tảng phổ biến dùng tạo nên các ứng dụng Web mạnh mẽ và giờ đây Crystal Reports đã trở thành một phần vận hành công việc trình bày báo biểu và kết xuất từ các ứng dụng Web. ASP.NET là công nghệ của Microsoft dành cho nền Windows. Nếu bạn sử dụng công nghệ Java thì chương sau sẽ hướng dẫn bạn cách tích hợp Crystal Reports vào trong ứng dụng JSP (Java Server Pages).

Chương 11:

TÍCH HỢP BÁO BIỂU VÀO ỨNG DỤNG JAVA

- ❖ *Tổng quan.*
- ❖ *Tạo báo biểu sử dụng với ứng dụng JSP.*
- ❖ *Làm việc với các thành phần báo biểu Java.*
- ❖ *Làm việc với các Field tham số.*
- ❖ *Tóm tắt.*

1. TỔNG QUAN

Crystal Reports đã trở nên quen thuộc với các nhà phát triển ứng dụng theo dòng sản phẩm Microsoft, nhưng nó ít được các chuyên gia ứng dụng Java biết đến. Đó là do Crystal Reports thường kết hợp với các sản phẩm của bộ công cụ Visual Studio. Tuy nhiên, trong vài năm trở lại đây, BusinessObjects đã giúp một số nhà cung cấp công cụ tích hợp phiên bản OEM của Crystal Reports với thành phần Java Report (Java Report Component) vào bộ công cụ phát triển Java như: BEA Weblogic Workshop, Borland® JBuilder, IBM Rational Application Developer, IBM WebSphere® Studio.

Việc mở rộng phạm vi này đã khiến Crystal Reports có khả năng hỗ trợ 100% mã Java. JRC (Java Report Component) được giới thiệu từ phiên bản Crystal Reports 10 đã tạo cơ hội cho các chuyên viên phát triển Java sử dụng Crystal Reports nhiều hơn trong ứng dụng của họ. JRC cung cấp các công cụ thiết lập mạnh mẽ cho việc tích hợp báo biểu Report vào ứng dụng của bạn với khả năng viết mã ít nhất.

Ghi chú: Do JRC thuần Java, nó có thể sử dụng với các nền phát triển Java và các Server ứng dụng như WebLogic, WebSphere và Tomcat.

Trong chương này chúng ta sẽ xem xét đến việc sử dụng JRC để thêm báo biểu Report vào ứng dụng JSP.

2. TẠO BÁO BIỂU SỬ DỤNG VỚI ỨNG DỤNG JSP

Phần sau sẽ hướng dẫn bạn kết nối nguồn dữ liệu để tạo ra báo biểu trong các ứng dụng JSP.

2.1. Tìm hiểu JDBC

Khi làm việc với Crystal Reports và Java, có hai cách khác nhau mà bạn có thể sử dụng để xuất dữ liệu trong báo biểu. Cách thứ nhất là sử dụng trình JDBC™ để kết nối trực tiếp vào các nguồn dữ liệu. Cách thứ hai là dùng cơ chế cầu nối ODBC-JDBC.

Đối với cả hai cách này, trước tiên bạn cài đặt và cấu hình JDBC driver theo yêu cầu của nhà sản xuất đưa ra. Tiếp đến, bạn cần phải cập nhật file CRConfig.XML, nằm trong thư mục C:\Program Files\Common Files\Business Objects\3.0\Java. Trong file XML này có nhiều cấu hình thiết lập khác nhau và bạn có thể tham khảo trong bảng dưới đây:

Nội dung	Mô tả	Ví dụ
JavaDir	Đường dẫn đến thư viện thực thi Java.	C:\Program Files\Java\jre1.5.0_02\bin
ClassPath	Đường dẫn trỏ đến file JAR chứa trình JDBC Driver.	
JDBCURL	Chuỗi URL mà JDBC driver sử dụng.	Jdbc:oracle:oci8:scott/tiger@myhost
JDBCUserName	Tên đăng nhập.	Scott
JDBCClassName	Lớp JDBC driver.	com.oracle.jdbc.oracle.oci8

Dưới đây là nội dung của file CRConfig.XML

```
<?xml version="1.0" encoding="utf-8"?>
<CrystalReportEngine-configuration>
  <reportlocation>../..</reportlocation>
  <timeout>10</timeout>
```

```

<ExternalFunctionLibraryClassNames>
    <classname> </classname>
    <classname> </classname>
</ExternalFunctionLibraryClassNames>

<keycode>XXXXXX-XXXXX-XXXX-XXXX</keycode>
<Javasever-configuration>
<DataDriverCommon>
<JavaDir>C:\Program Files\Java\jre1.5.0_02\bin</JavaDir>
  <Classpath>C:\Program Files\Common Files\Business
Objects\3.0\java\lib\crlovmanifest.jar;C:\Program Files\Common
Files\Business Objects\3.0\java\lib\CRLOVEexternal.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\CRDBJavaServerCommon.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\CRDBJavaServer.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\CRDBJDBCServer.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\CRDBXMLServer.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\CRDBJavaBeansServer.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\external\CRDBXMLExternal.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\external\log4j.jar;C:\Program Files\Common
Files\Business Objects\3.0\java\lib\cecore.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\celib.jar;C:\Program Files\Common
Files\Business Objects\3.0\java\lib\ebus405.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\corbaidl.jar;C:\Program Files\Common
Files\Business
Objects\3.0\java\lib\external\freessl201.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\external\asn1.jar;C:\Program Files\Common
Files\Business Objects\3.0\java\lib\external\certj.jar;C:\Program
Files\Common Files\Business
Objects\3.0\java\lib\external\jsafe.jar;C:\Program Files\Common
Files\Business Objects\3.0\java\lib\external\sslj.jar; C:\Program

```

```
Files\Microsoft SQL Server 2000 Driver for JDBC\lib\msbase.jar;
C:\Program Files\Microsoft SQL Server 2000 Driver for
JDBC\lib\mssqlserver.jar; C:\Program Files\Microsoft SQL Server
2000 Driver for JDBC\lib\msutil.jar;${CLASSPATH}</Classpath>
<IORFileLocation>${TEMP}</IORFileLocation>
<JavaServerTimeout>1800</JavaServerTimeout>
<JVMMaxHeap>64000000</JVMMaxHeap>
<JVMMinHeap>32000000</JVMMinHeap>
<NumberOfThreads>100</NumberOfThreads>
</DataDriverCommon>
```

<JDBC>

```
<CacheRowsetSize>100</CacheRowsetSize>
<JDBCURL>jdbc:microsoft:sqlserver://sqlserver01:1443
</JDBCURL>
<JDBCClassName>
com.microsoft.jdbc.sqlserver.SQLServerDriver
</JDBCClassName>
<JDBCUserName>sa</JDBCUserName>
<JNDIURL></JNDIURL>
<JNDIConnectionFactory></JNDIConnectionFactory>
<JNDIInitContext></JNDIInitContext>
<JNDIUserName>weblogic</JNDIUserName>
<GenericJDBCdriver>
<Option>No</Option>
<DatabaseStructure>catalogs,tables</DatabaseStructure>
<StoredProcType>Standard</StoredProcType>
<LogonStyle>SQLServer</LogonStyle>
</GenericJDBCdriver>
```

</JDBC>

<XML>

```
<CacheRowsetSize>100</CacheRowsetSize>
<PreReadNBytes>4096</PreReadNBytes>
<XMLLocalURL></XMLLocalURL>
<SchemaLocalURL></SchemaLocalURL>
<XMLHttpURL></XMLHttpURL>
<SchemaHttpURL></SchemaHttpURL>
```

</XML>

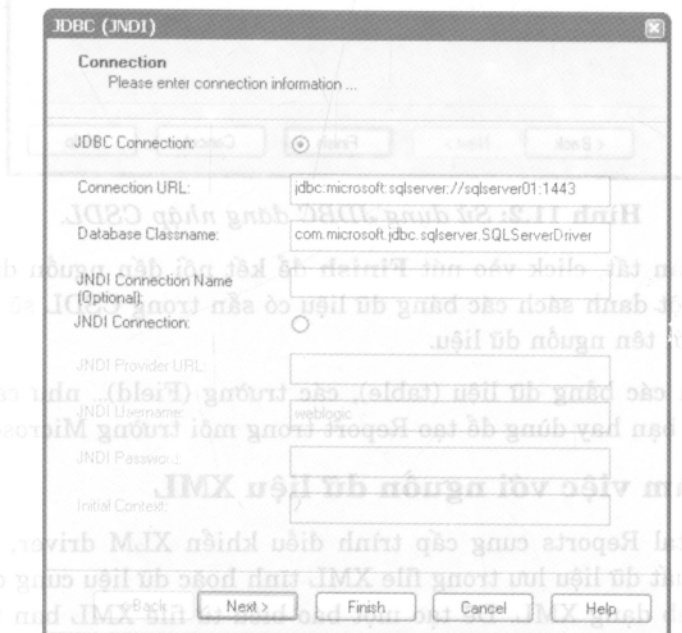
```

<JavaBeans>
  <CacheRowsetSize>100</CacheRowsetSize>
  <JavaBeansClassPath></JavaBeansClassPath>
</JavaBeans>
</Javaserver-configuration>
</CrystalReportEngine-configuration>

```

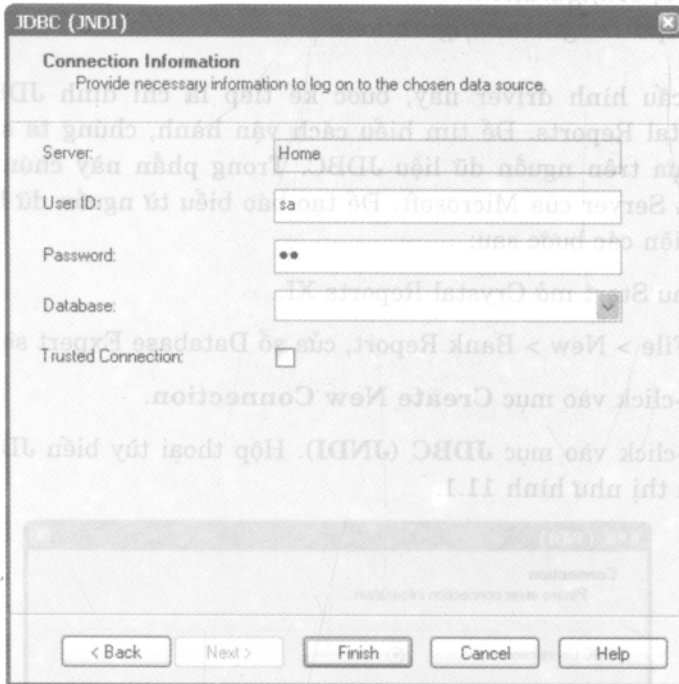
Với cấu hình driver này, bước kế tiếp là chỉ định JDBC driver trong Crystal Reports. Để tìm hiểu cách vận hành, chúng ta sẽ tạo một báo biểu dựa trên nguồn dữ liệu JDBC. Trong phần này chúng ta dùng CSDL SQL Server của Microsoft. Để tạo báo biểu từ nguồn dữ liệu JDBC bạn thực hiện các bước sau:

1. Từ Menu Start mở Crystal Reports XI.
2. Nhấp File > New > Bank Report, cửa sổ Database Expert sẽ hiển thị.
3. Double-click vào mục **Create New Connection**.
4. Double-click vào mục **JDBC (JNDI)**. Hộp thoại tùy biến JDBC driver sẽ hiển thị như hình 11.1.



Hình 11.1: JDBC driver.

5. Nhập đường dẫn kết nối vào ô **Connection URL** và lớp truy xuất CSDL vào **Database Classname**, sử dụng các giá trị giống như bạn đã nhập vào trong file CRConfig.XML, sau đó click **Next**.
6. Sử dụng hộp thoại kết nối như hình 11.2 để nhập tên người dùng (User Name) và mật khẩu (Password) cho CSDL của bạn.



Hình 11.2: Sử dụng JDBC đăng nhập CSDL.

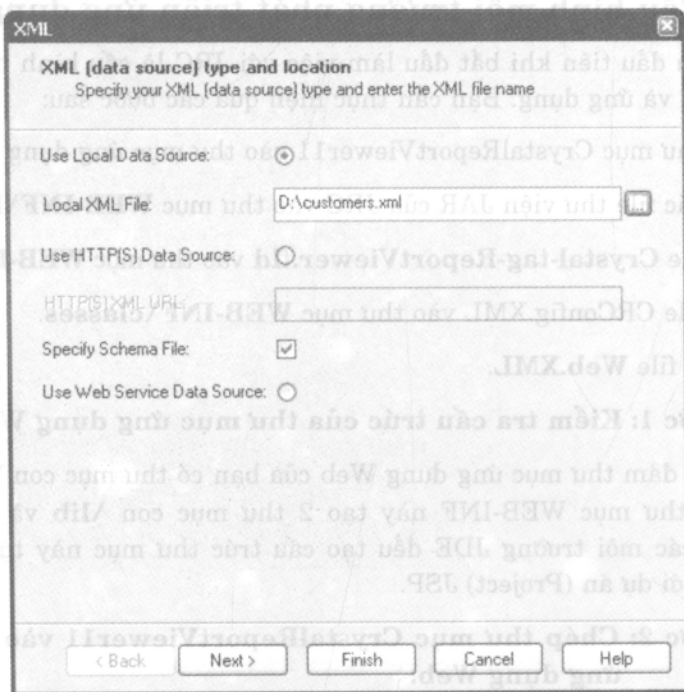
7. Khi hoàn tất, click vào nút **Finish** để kết nối đến nguồn dữ liệu của bạn. Một danh sách các bảng dữ liệu có sẵn trong CSDL sẽ xuất hiện bên dưới tên nguồn dữ liệu.

Chọn các bảng dữ liệu (table), các trường (Field)... như cách thông thường mà bạn hay dùng để tạo Report trong môi trường Microsoft.

2.2. Làm việc với nguồn dữ liệu XML

Crystal Reports cung cấp trình điều khiển XLM driver, cho phép bạn truy xuất dữ liệu lưu trong file XML tĩnh hoặc dữ liệu cung cấp từ các dịch vụ Web dạng XML. Để tạo một báo biểu từ file XML bạn thực hiện qua các bước sau:

1. Từ Menu Start mở Crystal Reports XI.
2. Nhấp **File > New > Bank Report**, cửa sổ Database Expert sẽ hiển thị.
3. Mở rộng mục **Create New Connection** và ở bên trong mục này mở rộng tiếp mục **XML**. Hộp thoại nguồn dữ liệu XML sẽ hiển thị như hình 11.3.



Hình 11.3: Driver dữ liệu XML.

4. Nhấp vào nút “...” trong hộp thoại để duyệt tìm file XML dữ liệu. Trong ví dụ này bạn có thể sử dụng file CUSTOMERS.XML nằm trên đĩa CD-ROM kèm giáo trình. Sau khi chọn file, click vào nút **Finish** để trở lại hộp thoại DataBase Expert.
5. File dữ liệu XML bây giờ sẽ xuất hiện bên dưới mục XML. Bạn có thể tạo báo biểu theo như cách tạo một báo biểu thông thường mà chúng ta vẫn làm (chọn trường, thêm nhóm, sắp xếp...).

Crystal Reports sẽ xem file XML như là một bảng dữ liệu và lấy ra kiểu dữ liệu thích hợp cho mỗi trường nó nhận diện được.

3. LÀM VIỆC VỚI CÁC THÀNH PHẦN BÁO BIỂU JAVA

JRC (Java Reporting Component) là thành phần giúp tích hợp các báo biểu vào trong các ứng dụng Java. Trong phần tiếp theo sau, chúng ta sẽ học cách cấu hình và sử dụng JRC để hiển thị nội dung báo biểu.

3.1. Cấu hình môi trường phát triển ứng dụng

Điều đầu tiên khi bắt đầu làm việc với JRC là cấu hình môi trường phát triển và ứng dụng. Bạn cần thực hiện qua các bước sau:

1. Chép thư mục CrystalReportViewer11 vào thư mục ứng dụng Web.
2. Chép các file thư viện JAR của JRC vào thư mục **WEB-INF\lib**.
3. Chép file **Crystal-tag-ReportViewer.tld** vào thư mục **WEB-INF\lib**.
4. Chép file CRConfig.XML vào thư mục **WEB-INF\classes**.
5. Sửa đổi file **Web.XML**.

Bước 1: Kiểm tra cấu trúc của thư mục ứng dụng Web:

Bảo đảm thư mục ứng dụng Web của bạn có thư mục con **WEB-INF** và trong thư mục WEB-INF này tạo 2 thư mục con **\lib** và **\classes**. Hầu hết các môi trường JDE đều tạo cấu trúc thư mục này tự động khi bạn tạo mới dự án (Project) JSP.

Bước 2: Chép thư mục CrystalReportViewer11 vào thư mục ứng dụng Web:

Thư mục này chứa tất cả file cần thiết cho trình xem báo biểu. Bạn có thể tìm thấy thư mục này theo đường dẫn C:\Program Files\Business Objects\Common\3.0\CrystalReportViewers11.

Bước 3: Chép thư mục Java Report Component JAR vào thư mục WEB-INF\lib:

Các file JAR cho JRC nằm trong C:\Program Files\Business Objects\Common\3.0\java\lib. Bạn cần chép tất cả các file này vào thư mục con **\lib** của ứng dụng Web. Kiểm tra rằng bạn đã chép tất cả file JAR có trong thư mục con để đảm bảo bạn có tất cả các thành phần của nhà cung cấp thứ ba mà JRE (Java Runtime Environment) cần.

Bước 4: Chép Crystal-tags-ReportViewer.tld vào thư mục WEB-INF\lib của bạn:

File này được yêu cầu mỗi khi bạn sử dụng trình xem tag (viewer tag) Crystal Reports. Bạn có thể tìm thấy file theo đường dẫn C:\Program Files\Business Objects\Common\3.0\java\lib\taglib. Để có thêm thông tin về cách sử dụng Report viewer tags, bạn xem thêm tài liệu Java SDK đi kèm theo với Crystal Reports XI.

Bước 5: Chép file CRConfig.XML vào thư mục WEB-INF\classes:

File này chứa cấu hình tùy biến cho Report engine, hãy chép nó vào thư mục con \classes.

Bước 6: Sửa đổi file Web.xml của bạn:

Bước cuối cùng là sửa đổi file Web.xml của ứng dụng. Bạn sẽ cần thêm vào đoạn cấu hình sau:

```
<context-param>
  <Para-name>crystal_image_uri</Para-name>
  <Para-value>crystalreportviewers11</Para-value>
</context-param>
```

Và nếu có dự định sử dụng thư viện Crystal Viewer tag, bạn cần thêm vào đoạn cấu hình sau:

```
<taglib>
  <taglib-url>
    /crystal-tags-reportviewer.tld
  </taglib-url>
  <taglib-location>
    /WEB-INF/crystal-tags-reportviewer.tld
  </taglib-location>
</taglib>
```

Chỉ khi nào đã hoàn tất những bước này, bạn mới có thể sử dụng tích hợp báo biểu với JRCI vào ứng dụng JSP của mình.

3.2. Xem báo biểu trên trang Web

Để xem một báo biểu bằng JRC, bạn sử dụng lớp `CrystalReportViewer` do JRC cung cấp. Tương tự với thành phần `CrystalReportViewer` của Visual Studio, thành phần `CrystalReportViewer` của JRC cũng chứa tất cả các thuộc tính và phương thức dùng cho việc hiển thị báo biểu và điều khiển nội dung báo biểu.

Trước hết, để có thể xem Report từ trang JSP, bạn cần tạo một tham chiếu lớp ở đầu trang JSP như sau:

```
<%@page import=
    "com.crystaldecisions.report.web.viewer.*"%>
<%@page import ="com.crystaldecisions.report.sdk.*"%>
```

Sau đó tạo một đối tượng thuộc lớp `ReportClientDocument` và mở file báo biểu:

```
String MyReportName = "WEBSALES.rpt";
ReportClientDocument myReportClientDocument =
    new ReportClientDocument();
myReportClientDocument.Open(MyReportName, 0);
```

Bước kế tiếp để xây dựng nội dung báo biểu là lấy về đối tượng `ReportSource` bằng đoạn mã sau:

```
Object MyReportSource;
MyReportSource =
```

```
myReportClientDocument.getReportSource();
```

Tiếp đến là đặt nội dung báo biểu vào Viewer để hiển thị:

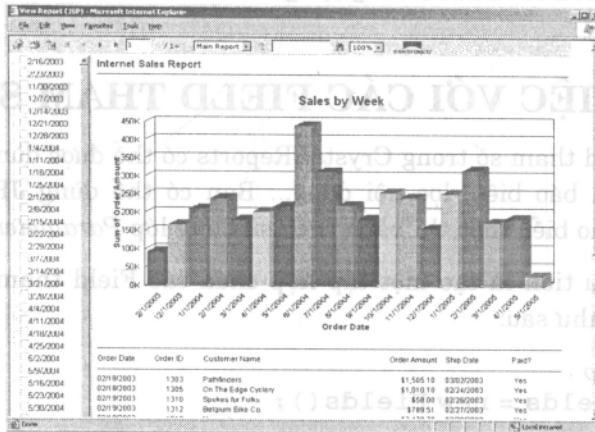
```
CrystalReportViewer myCrystalReportViewer =
    new CrystalReportViewer();
myCrystalReportViewer.SetReportSource(
    MyReportSource);
myCrystalReportViewer.setOwnPage(true);
myCrystalReportViewer.setOwnForm(true);
```

Cuối cùng, chúng ta cần làm tươi báo biểu để đọc dữ liệu sau cùng nhất và dùng phương thức `processHttpRequest` để xử lý yêu cầu hiển thị báo biểu.

```
myCrystalReportViewer.refresh();
myCrystalReportViewer.processHttpRequest(request,
```

```
response, getServletConfig().getServletContext(),
null);
```

Khi biên dịch và chạy trang ứng dụng, Report của bạn sẽ được hiển thị bằng trình duyệt HTML cung cấp bởi Crystal Reports XI và sẽ trông như hình 11.4.



Hình 11.4: Xem báo biểu trên HTML.

Nếu bạn không thể biên dịch và chạy ứng dụng, hãy kiểm tra lại các bước và xem Report của bạn nằm đúng vị trí đường dẫn hay chưa. Dưới đây là toàn bộ mã nguồn của trang JSP.

Viewreport.JSP

```
<%@Page import=
    "com.crystaldecisions.report.web.viewer.*"%>
<%@Page import="com.crystaldecisions.report.sdk.*"%>
<%
String MyReportName = "WEBSALES.rpt";
ReportClientDocument myReportClientDocument = new
CrystalReportDocument();
myReportClientDocument.open(MyReportName, 0);

Object MyReportSource;
MyReportSource =
myReportClientDocument.getReportSource();

CrystalReportViewer myCrystalReportViewer = new
CrystalReportViewer();
myCrystalReportViewer.setReportSource(
    MyReportSource);
```

```
myCrystalReportViewer.SetOwnPage(true);
myCrystalReportViewer.SetOwnForm(true);

myCrystalreportViewer.refresh();
myCrystalreportViewer.processHttpRequest(request,
response, getServletConfig().getServletContext(), null);
%>
```

4. LÀM VIỆC VỚI CÁC FIELD THAM SỐ

Các Field tham số trong Crystal Reports có thể được dùng để hiển thị thông tin trên báo biểu, lọc nội dung... Bạn có thể dùng JRC để truyền tham số khi báo biểu thực thi bằng cách sử dụng lớp *ParameterField*.

Bước đầu tiên là tạo một tập hợp chứa các Field tham số bằng mã chương trình như sau:

```
// Tạo tập hợp .
Fields MyFields = new Fields();

// Tạo một giá trị biến mới và thiết lập nó là "USA"
String MyParameterValue = "USA"
```

Tiếp theo là tạo Field tham số và đưa nó vào tập hợp

```
// Tạo đối tượng Field tham số
ParameterField MyParameterField = new ParameterField();
MyParameterField.setReportName(MyReportName);
Values MyParameterValues = new Values();

// Tạo giá trị Field tham số
ParameterFieldDiscreteValue
    MyParameterFieldDiscreteValue =
        new ParameterFieldDiscreteValue();

// Thiết lập tên của biến và giá trị biến
MyParameterField.setName("?EnterCountry");
MyParameterFieldDiscreteValue.setValue(
    MyParameterValue);
MyParameterValues.add(MyParameterFieldDiscreteValue);
MyParameterField.setCurrentValues(MyParameterValues);
MyFields.add(MyParameterField);
```

Phần còn lại, khi thực thi báo biểu tham số sẽ lọc ra tất cả các mẫu tin có giá trị mã quốc gia là "USA".

5. ĐĂNG NHẬP TỰ ĐỘNG VÀO CSDL

Khi bạn tạo một báo biểu từ CSDL hoặc từ một nguồn dữ liệu khác, bạn phải nhập tên người dùng (username) và mật khẩu (password) để truy cập vào CSDL hoặc nguồn dữ liệu đó. Khi báo biểu chạy trong ứng dụng Web, để tránh người dùng phải nhập vào thông tin đăng nhập, bạn sử dụng đối tượng *ConnectionInfo* để cung cấp thông tin truy cập CSDL trước khi xem báo biểu. Trong ví dụ sau, tên tài khoản đăng nhập là "sa" và tên mật khẩu "admin" được truyền qua đối tượng *ConnectionInfo* sẽ khiến quá trình đăng nhập tự động, người sử dụng sẽ không còn bị nhắc nhở nhập giá trị kết nối này khi xem báo biểu.

```
// Thiết lập thông tin kết nối
ConnectionInfos MyConnectionInfos = new
    ConnectionInfos();
ConnectionInfo MyConnectionInfo = new ConnectionInfo();

// Thiết lập tên và mật khẩu CSDL
MyConnectionInfo.SetUserName("sa");
MyConnectionInfo.SetPassword("admin");
MyConnectionInfos.add(MyConnectionInfo);
```

Dưới đây là mã trang JSP thực hiện cơ chế kết nối tự động hiển thị thông tin báo biểu:

```
SETDB.JSP
<%@ Page import="com.crystaldecisions.report.sdk.*"%>
<%@ Page import=
    "com.crystaldecisions.sdk.occa.
        report.reportsource.*"%>
<%@ Page import=
    "com.crystaldecisions.sdk.occa.report.lib.*"%>
<%@ Page import=
    "com.crystaldecisions.sdk.occa.report.data.*"%>
<%

    // Mở tài liệu Report
    String MyReportName = "setdb.rpt"
    ReportClientDocument MyReportClientDocument = new
        ReportClientDocument();
    MyReportClientDocument.open(MyReportDocument);

    // Lấy Report
    Object MyReportSource;
    MyReportSource =
        MyReportClientDocument.getReportSource();
```

```
// Thiết lập thông tin kết nối
ConnectionInfos MyConnectionInfos = new
    ConnectionInfos();
ConnectionInfo MyConnectionInfo = new ConnectionInfo();

// Thiết lập tên và mật khẩu CSDL
MyConnectionInfo.setUserName("sa");
MyConnectionInfo.setPassword("admin");
MyConnectionInfos.add(MyConnectionInfo);

// Tạo viewer và thiết lập report source cho Report
CrystalReportViewer MyCrystalReportViewer = new
    CrystalReportViewer();
MyCrystalReportViewer.setReportSource(MyReportSource);

// Thiết lập tùy biến viewer
MyCrystalReportViewer.setOwnpage(true);
MyCrystalReportViewer.setOwnForm(true);
MyCrystalReportViewer.setDatabaseLogonInfos(
    MyConnectionInfos);

// Làm tươi Report
MyCrystalReportViewer.refresh();
MyCrystalReportViewer.processHttpRequest(
    request, response,
    getServletConfig().getServletContext(), null);
%>
```

6. TÓM TẮT

Java Server Pages (JSP) là công cụ phát triển ứng dụng Web mạnh mẽ. Giờ đây với sự hỗ trợ của Crystal Reports, bạn có thể nhúng mã xử lý báo biểu hoàn toàn bằng Java, tận dụng sức mạnh của Crystal vốn là lợi thế của các nhà phát triển trên nền Microsoft. Trong chương tiếp theo, chúng ta sẽ học cách tích hợp báo biểu vào ứng dụng cao cấp sử dụng BusinessObjects Enterprise XI.

Chương 12:

BUSINESSOBJECTS ENTERPRISE

- ❖ *Tổng quan.*
- ❖ *Tìm hiểu kiến trúc Report Server.*
- ❖ *Cài đặt BusinessObjects.*
- ❖ *Làm việc xử lý báo biểu với đường dẫn URL.*
- ❖ *Xem báo biểu.*
- ❖ *Tóm tắt.*

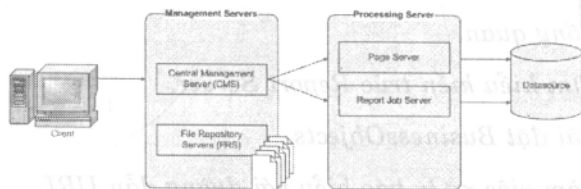
1. TỔNG QUAN

Trong một ứng dụng lớn có hàng ngàn báo biểu và truy xuất bởi nhiều người với các quyền truy cập khác nhau, bạn chắc chắn sẽ cần đến kiến trúc BusinessObjects Enterprise. Đây là kiến trúc quản lý báo biểu phía máy chủ Server hỗ trợ lập trình phân tán cho những hệ thống ứng dụng lớn. BusinessObjects Enterprise cung cấp:

- ♦ Kiến trúc mạnh mẽ cho việc đăng nhập và xử lý Report trên nhiều máy chủ Server.
- ♦ Kỹ thuật lưu trữ Report và khả năng xem trên trình duyệt Web.
- ♦ Engine lập kế hoạch chạy báo biểu định kỳ linh động theo một khoảng thời gian và ngày cụ thể.
- ♦ Tùy biến phân tán, chuyển tải, lưu trữ báo biểu theo email, ra máy in, đĩa (disk), FTP.
- ♦ Khả năng xuất báo biểu ra nhiều định dạng khác nhau như: Adobe Acrobat Reader, Excel, Word, RTF, Text.
- ♦ Quản lý bảo mật báo biểu cho phép nhiều người truy cập.

2. TÌM HIỂU KIẾN TRÚC SERVER

BusinessObjects Enterprise cung cấp một kiến trúc đa Server, mỗi Server xử lý những tác vụ hoàn toàn độc lập. Hình 12.1 là cấu trúc cơ bản của BusinessObjects Enterprise.



Hình 12.1: Kiến trúc cơ bản của BusinessObjects Enterprise.

Máy chủ trung tâm gọi là CMS (Central Management Server). CMS tiếp nhận xử lý các yêu cầu gửi đến từ các máy chủ Web Server như các tác vụ đăng nhập, xác thực quyền xem báo biểu, lập lịch thực thi Report. Các báo biểu Report muốn sử dụng trong môi trường BusinessObjects đều phải được lưu vào kho chứa (quá trình này gọi là xuất bản hay Publishing) theo cách sau: trong môi trường thiết kế Crystal Reports, chọn **File > Save As** và nhấp vào biểu tượng Enterprise để đăng nhập vào BusinessObjects Enterprise và lưu Report vào thư mục riêng. Bạn cũng có thể sử dụng công cụ Crystal Management Console, hoặc Web Publishing Wizard của Crystal để xuất bản file Report lưu vào kho chứa của BusinessObjects.

Sau khi xuất bản Report, một mục nhập chứa thông tin báo biểu sẽ được lưu vào hệ thống CSDL của CMS và báo biểu được sao chép một bản vào máy chủ lưu trữ file mang tên FRS (File Repository Server). FRS có thể phân làm hai loại, máy chủ FRS “nguồn” (hay input) sẽ chứa tất cả các mẫu template và máy chủ khác gọi là FRS “kết xuất” (hay output) sẽ lưu nội dung báo biểu kết quả sau khi thực thi. Các Report template chứa trên FRS nguồn có thể lập lịch biểu để chạy thực thi cho ra các Report lưu vào FRS “kết quả” chờ người dùng yêu cầu hiển thị.

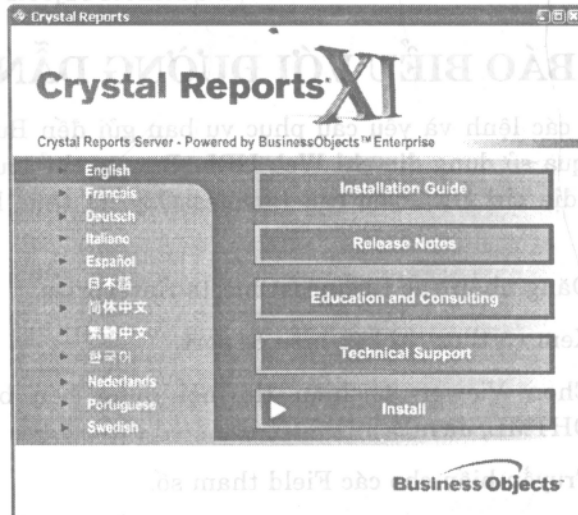
3. CÀI ĐẶT BUSINESSOBJECTS ENTERPRISE

XI

Để tiến hành cài đặt BusinessObjects Enterprise XI, chạy file *Setup.exe* từ đĩa CD-ROM. Cửa sổ chào mừng hiển thị như hình minh họa 12.2.

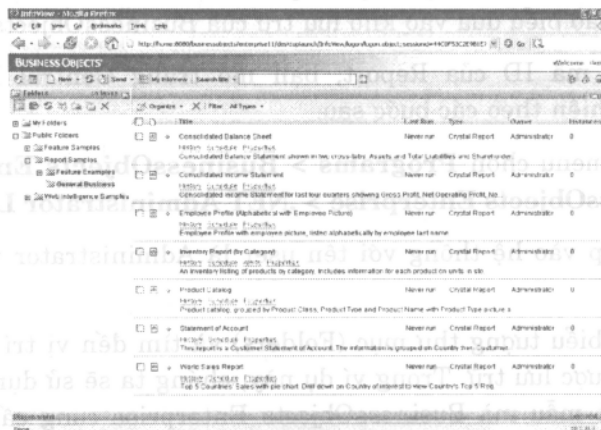
Chọn Install, trình Wizard sẽ hướng dẫn bạn qua các bước cài đặt, bạn chỉ việc nhấn nút Next cho đến khi quá trình cài đặt hoàn tất.

Theo mặc định, BusinessObjects Enterprise XI được cài đặt với quyền quản trị (Administrator) và mật khẩu trống.



Hình 12.2: Giao diện cài đặt BusinessObjects Enterprise XI.

Bạn có thể kiểm tra quá trình cài đặt diễn ra thành công hay không sau đó bằng cách sử dụng Central Management Console. Để gọi trình này bạn mở menu **Start** và click vào **Programs > BusinessObjects Enterprise XI > .NET Administrator Launchpad**.



Hình 12.3: Crystal Management Console.

Một trang Web sẽ hiển thị. Bạn có thể đăng nhập vào trong Crystal Management Console bằng tài khoản Administrator chỉ định trong quá trình cài đặt. Sau khi đăng nhập bạn có thể quản lý người dùng (User), nhóm người dùng (Group), thư mục (Folder), xuất bản báo biểu Report của mình lưu vào kho chứa của BusinessObjects Enterprise.

4. XỬ LÝ BÁO BIỂU VỚI ĐƯỜNG DẪN URL

Hầu hết các lệnh và yêu cầu phục vụ bạn gửi đến BusinessObjects Server thông qua sử dụng địa chỉ Web URL. Bạn có thể truyền các lệnh, tham số trên địa chỉ URL yêu cầu BusinessObjects thực hiện các công việc như:

- Đăng nhập vào BusinessObjects Enterprise.
- Xem và thực thi báo biểu Report.
- Chọn Viewer để hiển thị nội dung báo biểu (ActiveX, DHTML, Java).
- Truyền biến cho các Field tham số.
- Thiết lập công thức lọc mẫu tin.
- Chuyển đổi định dạng báo biểu Report.

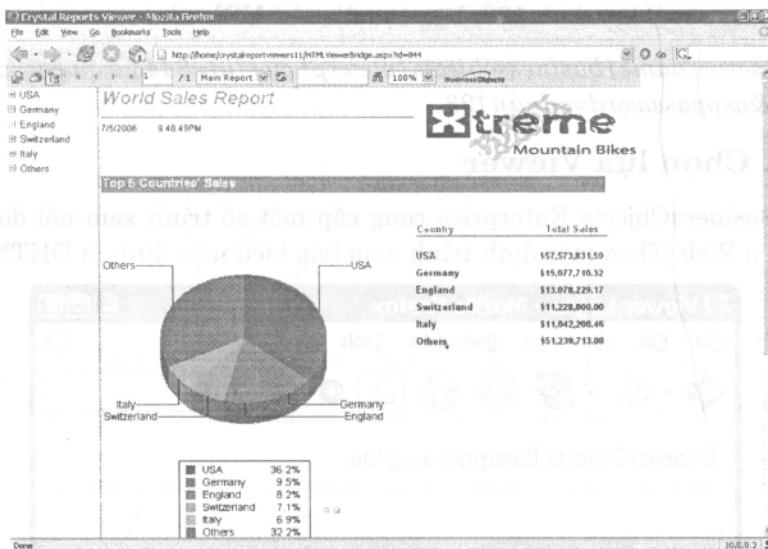
4.1. Xem báo biểu

Muốn xem báo biểu nào đó trong BusinessObjects bạn phải biết mã số ID của nó. Mã số này được cấp khi lần đầu tiên bạn xuất bản (Publishing) báo biểu đưa vào kho lưu trữ của BusinessObjects.

Để tìm mã ID của Report, bạn mở trang Central Management Console thực hiện theo các bước sau:

1. Từ Start menu chọn **Programs > BusinessObjects Enterprise 11 > BusinessObjects Enterprise > .NET Administrator Launchpad**.
2. Đăng nhập vào hệ thống với tên user là Administrator và mật khẩu để trống.
3. Click vào biểu tượng thư mục (Folders) và tìm đến vị trí thư mục nơi báo biểu được lưu trữ. Trong ví dụ này, chúng ta sẽ sử dụng một trong các Report mẫu mà BusinessObjects Enterprise cung cấp. Bạn chọn vị trí **Report Samples > General Business**, sau đó chọn Report **World Sales**.

4. Click vào Report World Sales để mở trang thuộc tính của nó và nhấn nút Preview để xem nội dung báo biểu Report.



Hình 12.4: Xem bản biểu World Sales từ URL.

5. Hãy nhìn vào đường dẫn URL trên thanh Address Bar của trình duyệt Web, tại đây bạn sẽ thấy mã số ID của báo biểu do BusinessObject sinh ra. Ví dụ, trên máy chủ của chúng tôi mã số ID là 844 nhưng nó có thể khác trên Server của bạn.

Để xem một báo biểu Report sử dụng địa chỉ URL, bạn gán ID của Report theo cú pháp sau:

`http://servername/businessobjects/viewrpt.cwr?id=Mã số ID`

Ví dụ:

`http://servername/businessobjects/viewrpt.cwr?id=844`

4.2. Đăng nhập vào BusinessObjects Enterprise

Mỗi khi sử dụng địa chỉ Web URL để xem báo biểu Report, bạn sẽ bị nhắc nhở nhập vào tài khoản và mật khẩu thông qua trang Log On như hình 12.5. Muốn đăng nhập tự động, bạn cần sử dụng thêm các tham số sau trên địa chỉ URL:

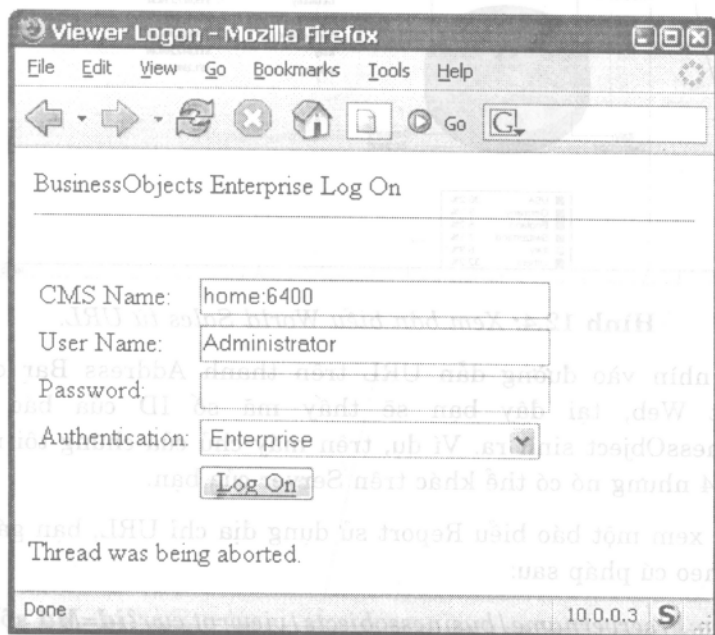
- APSUSER
- APSPASSWORD

Các tham số phân cách nhau bằng dấu "&". Ví dụ để mở báo biểu có mã số ID là 844 theo tên đăng nhập (username) là Administrator, mật khẩu (password) là admin123, bạn tạo địa chỉ URL như sau:

`http://servername/businessobjects/viewrpt.cwr?id=844&aspuser=admini
strator&asppassword=admin123`

4.3. Chọn lựa Viewer

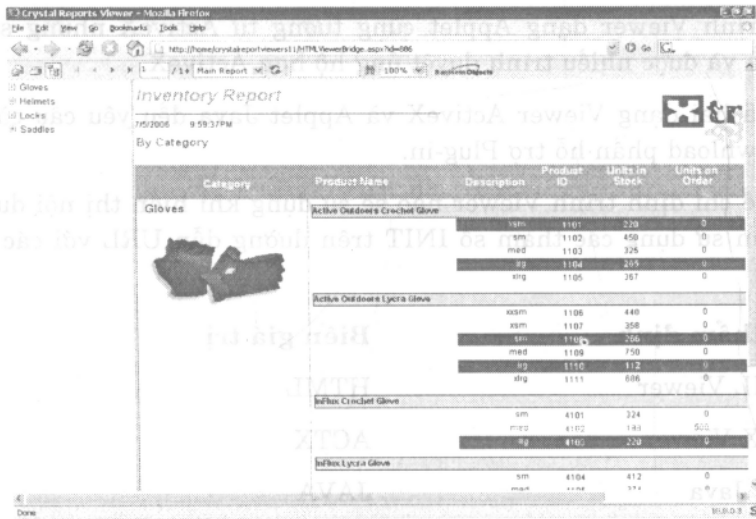
BusinessObjects Enterprise cung cấp một số trình xem nội dung báo biểu trên Web. Theo mặc định trình xem báo biểu mặc định là DHTML.



Hình 12.5: DHTML Viewer.

Trình xem báo biểu DHTML có lợi thế là xem được trên hầu hết các trình duyệt mà không cần cài thêm bất kỳ dạng hỗ trợ plug-in nào. Nó cung cấp phần lớn các chức năng mà người xem Report cần đến (Ví dụ: hộp danh sách sổ xuống, tìm kiếm trong báo biểu...).

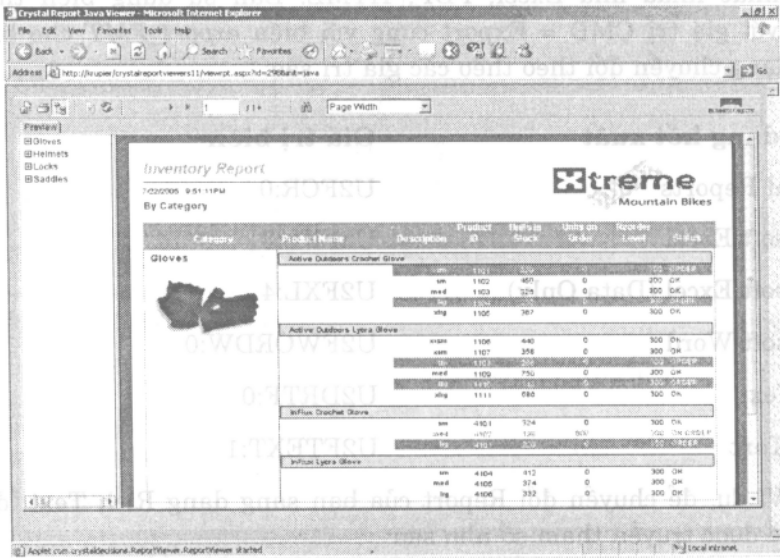
Cách xem báo biểu thứ hai trong BusinessObjects Enterprise là sử dụng Viewer dạng điều khiển ActiveX nhưng trong trang Web như hình 12.6.



Hình 12.6: ActiveX Viewer.

Trình Viewer ActiveX chỉ làm việc với trình duyệt Internet Explorer. Nó có giao diện và thao tác hoàn toàn giống với Viewer của ứng dụng Windows Desktop.

Cách xem báo biểu cuối cùng là sử dụng Applet của Java như hình 12.7.



Hình 12.7: Java Viewer.

Trình Viewer dạng Applet cũng tương tự ActiveX nhưng sử dụng mã Java và được nhiều trình duyệt ủng hộ hơn ActiveX.

Cả hai dạng Viewer ActiveX và Applet Java đều yêu cầu cài thêm hoặc download phần hỗ trợ Plug-in.

Để chỉ định trình Viewer nào sẽ sử dụng khi hiển thị nội dung báo biểu, bạn sử dụng các tham số INIT trên đường dẫn URL với các giá trị như sau:

Kiểu thẩm định	Biến giá trị
DHTML Viewer	HTML
ActiveX Viewer	ACTX
Applet Java	JAVA

Ví dụ, để xem báo biểu Report bằng Viewer ActiveX, bạn truyền tham số trên địa chỉ như sau:

<http://servername/businessobjects/viewrpt.cwr?id=844&init=actx>

4.4. Chuyển đổi định dạng báo biểu

Bạn có thể yêu cầu chuyển đổi nội dung báo biểu thành các định dạng khác nhau như Excel, PDF, HTML. Bạn sử dụng biến tham số CMD với giá trị CMD = Export cùng với biến *export_FMT* cho biết các định dạng chuyển đổi theo các giá trị sau:

Định dạng kết xuất	Giá trị biến
Crystal Reports	U2FCR:0
Microsoft Excel	U2FXL:3
Microsoft Excel (Data Only)	U2FXL:4
Microsoft Word	U2FWORDW:0
Rich Text	U2DRTF:0
Plain Text	U2FTEXT:1

Ví dụ, để chuyển đổi Report của bạn sang dạng Rich Text, địa chỉ URL sẽ được truyền tham số như sau:

http://servername/businessobjects/viewrpt.cwr?id=844&cmd=export&export_fmt=U2DRTF:0

Ngoài ra, bạn có thể chỉ định các trang thông qua biến tham số *export_OPT*. Ví dụ, để chuyển đổi trang 1-5 của báo biểu sang dạng RTF bạn xây dựng URL như sau:

[http://servername/businessobjects/viewrpt.cwr?id=844&cmd=export&export_fmt=U2DRTF:0&EXPORT_OPT=\[1-5\]](http://servername/businessobjects/viewrpt.cwr?id=844&cmd=export&export_fmt=U2DRTF:0&EXPORT_OPT=[1-5])

4.5. Làm việc với các Field tham số

Bạn có thể truyền các Field tham số của báo biểu trên đường dẫn URL thông qua chuỗi **PROMPTEX** kết hợp với tên Field tham số của báo biểu. Ví dụ, nếu bạn có một báo biểu Report chứa Field tham số *EnterCountry*, bạn có thể truyền giá trị của *EnterCountry* trên đường dẫn URL bằng cách đặt nối tên tham số và chuỗi PROMPTEX với dấu gạch ngang (*PROMPTEX-EnterCountry*) như sau:

[http://servername/businessobjects/viewrpt.cwr?id=844&promptex-entercountry="Australia"](http://servername/businessobjects/viewrpt.cwr?id=844&promptex-entercountry=)

Trong ví dụ này chúng ta truyền giá trị chuỗi cho Field tham số, tuy nhiên bạn hoàn toàn có thể truyền các kiểu giá trị khác, chẳng hạn nếu muốn truyền dạng số bạn có thể đặt giá trị trực tiếp không bọc trong dấu nháy như sau:

<http://servername/businessobjects/viewrpt.cwr?id=844&promptex-number=1500>

Bạn cũng có thể truyền biến dạng ngày bằng cách chỉ định hàm DATE theo khuôn dạng DATE(YYYY,MM,DD). Ví dụ để chỉ định ngày cho Field tham số mang tên *OrderDate*, địa chỉ URL sẽ được xây dựng như sau:

[http://servername/businessobjects/viewrpt.cwr?id=844&promptex-orderdate="Date\(2006,01,01\)"](http://servername/businessobjects/viewrpt.cwr?id=844&promptex-orderdate=)

Nếu Report của bạn chứa nhiều Field tham số, bạn có thể phân cách các tham số bằng ký hiệu (&) như sau:

[http://servername/businessobjects/viewrpt.cwr?id=844&promptex-orderdate="Date\(2006,01,01\)"&promptex-ordercount=10000](http://servername/businessobjects/viewrpt.cwr?id=844&promptex-orderdate=)

Để truyền dãy các giá trị cho một biến tham số bạn phân cách các giá trị bằng dấu phẩy (.). Ví dụ, chúng ta có thể truyền các giá trị *Australia*, *USA* và *Canada* cho biến tham số *EnterCountry* như sau:

```
http://servername/businessobjects/viewrpt.cwr?id=844&promptx-  
entrycountry="Australia","USA","Canada"
```

Với các giá trị liên tục bạn có thể chỉ định giá trị biên trên và biên dưới trong cặp ngoặc vuông như sau:

```
http://servername/businessobjects/viewrpt.cwr?id=844&promptx-  
orderamount=[10000-20000]
```

Nếu không lấy giá trị 10000 và 20000 bạn sử dụng dấu ngoặc đơn thay cho ngoặc vuông như sau:

```
http://servername/businessobjects/viewrpt.cwr?id=844&promptx-  
orderamount=(10000-20000)
```

Còn nếu lấy một giá trị biên trên/hay dưới bạn sử dụng dấu ngoặc vuông đặt phía giá trị muốn lấy, chẳng hạn để lấy giá trị 1000:

```
http://servername/businessobjects/viewrpt.cwr?id=844&promptx-  
orderamount=[10000-20000]
```

4.6. Lọc mẫu tin

Một chức năng hoạt động khác của URL là có thể thiết lập các giá trị lọc mẫu tin. Công thức lọc trong của báo biểu Report được dùng để trả về chỉ những mẫu tin thỏa điều kiện yêu cầu.

Bạn dùng biến lệnh SF để truyền công thức lọc này cho địa chỉ URL. Chẳng hạn nếu muốn lọc tất cả những khách hàng thuộc quốc gia *Australia* bạn có thể truyền công thức SF trên URL như sau:

```
ttp://servername/businessobjects/viewrpt.cwr?id=844&sf={Customer.  
Country}="Australia"
```

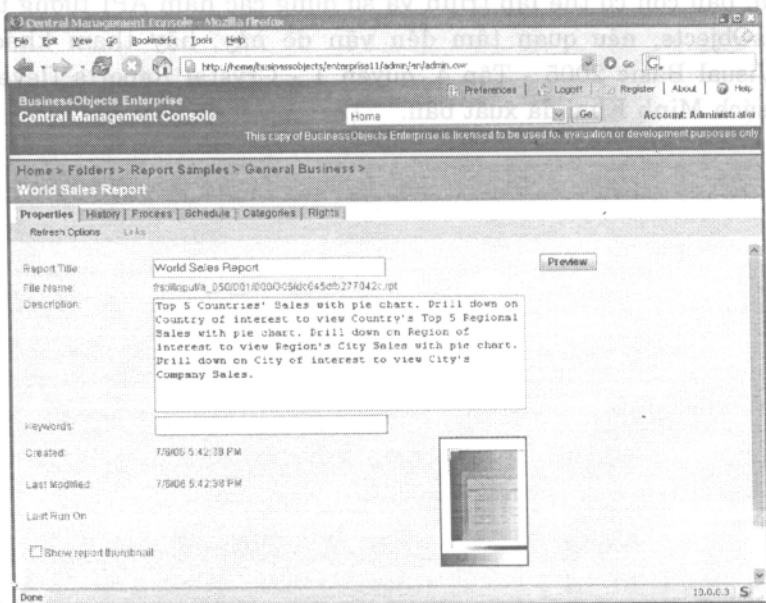
Bạn cũng có thể xây dựng một công thức lọc phức tạp hơn như ví dụ sau:

```
http://servername/businessobjects/viewrpt.cwr?id=844&sf=  
{Customer.Region}="New South Wales" and {Customer.City}="Sydney"  
and {Customer.Country}="Australia" and {Orders.Order Amount} >  
$10000.00 and {Orders.Order Date} > DateTime(2005, 01, 01, 00, 00, 00)
```

4.7. Làm việc với CSDL

Hầu hết các báo biểu Report đều dựa trên nguồn dữ liệu bảo mật, mỗi khi đọc nguồn dữ liệu này người sử dụng phải nhập vào tên (username) và mật khẩu (password) để xác thực quyền truy xuất CSDL. Bạn có thể chỉ định trước các thông tin đăng nhập để không phải chỉ định các thông tin này trên địa chỉ URL. Để làm điều này, bạn sử dụng Central Management Console và thực hiện theo các bước sau:

1. Từ *Start Menu*, click vào **Programs > BusinessObjects 11 > BusinessObjects Enterprise > .NET Administration Launchpad**.
2. Click vào đường link đến **Central Management Console**.
3. Đăng nhập vào **Central Management Console** với tên người dùng là Administrator và mật khẩu để trống.
4. Click vào biểu tượng Folder và di chuyển đến Folder nơi mà Report của bạn được lưu giữ.
5. Click vào tên Report để mở trang thuộc tính của nó, như hình 12.8.
6. Click vào tab Process sau đó click vào link Database. Trang thuộc tính Database sẽ hiển thị.



Hình 12.8: Trang thuộc tính báo biểu.

7. Nhập thông tin đăng nhập username và mật khẩu (password) vào để sử dụng cho Report này.
8. Cuộn xuống dưới cho đến cuối trang và chọn đánh dấu nút *Use Same Database Logon as When Report is run.*
9. Click vào nút *Update* ở phía cuối trang để lưu lại các thiết lập cho báo biểu.

Lúc này mỗi khi thực thi báo biểu bạn không còn cần phải nhập vào thông tin đăng nhập CSDL nữa.

5. TÓM TẮT

Kiến trúc BusinessObjects Enterprise phục vụ cho các ứng dụng lớn với hàng trăm báo biểu, khả năng quản lý Report mạnh mẽ, phân chia trên các máy chủ an toàn và bảo mật cao. Môi trường BusinessObjects cung cấp hầu như tất cả những cơ chế phân quyền, bảo vệ Report, xử lý các Report phức tạp với khối lượng dữ liệu kết xuất lớn, thực thi Report theo lịch định kỳ, điều mà những chương trình bình thường khó thực hiện được. Bạn chỉ chuyên tâm vào phát triển nội dung của báo biểu theo nghiệp vụ của mình, phần còn lại hãy để cho BusinessObjects lo liệu. Ngoài ra, bạn còn có thể lập trình và sử dụng các hàm API tương tác với BusinessObjects, nếu quan tâm đến vấn đề này, hãy tham khảo giáo trình “Visual Basic 2005 - Tập 4, quyển 1 - Crystal Reports Developer” do nhà sách Minh Khai đã xuất bản.

Sử dụng

CRYSTAL REPORTS XI

Thực hiện liên doanh: Công ty TNHH Minh Khai S.G
E-mail: mk.book@minhkhai.com.vn - Website: www.minhkhai.com.vn

Tổng phát hành

- ❖ Nhà sách Minh Khai: 249 Nguyễn Thị Minh Khai - Quận 1 - TP.HCM
ĐT: (08) 9.250.590 - 9.250.591 - Fax: (08) 9.257.837
- ❖ Nhà sách Minh Châu: Nhà 30 - Ngõ 22 - Tạ Quang Bửu - Bách Khoa - Hà Nội
ĐT: (04) 8.692.785 - Fax: (04) 8.683.995

Đại lý các khu vực

- ❖ Nhà sách Huy Hoàng: 95 Núi Trúc - Kim Mã - Ba Đình - Hà Nội
ĐT: (04) 7.365.859
 - ❖ Cty có phần sách thiết bị trường học Đà Nẵng: 78 Bạch Đằng - Đà Nẵng
ĐT: 0511.337100
 - ❖ Nhà sách Chánh Trí: 116A Nguyễn Chí Thanh - Đà Nẵng
ĐT: 0511.820129
 - ❖ Cty phát hành sách Khánh Hòa:
 - Nhà sách Ponagar: 73 Thống Nhất - Nha Trang - Khánh Hòa
ĐT: 058.822636
 - Siêu thị sách Tân Tiến: 11 Lê Thành Phương - Nha Trang - Khánh Hòa
ĐT: 058.827303
 - ❖ Nhà sách Năm Hiền: 79/6 Xô Viết Nghệ Tĩnh - TP. Cần Thơ
ĐT: 071. 821668
-

■ SỬ DỤNG CRYSTAL REPORTS XI

(Tin học và đời sống)

Chịu trách nhiệm xuất bản: Quang Thắng

Biên tập nội dung: Thảo Nguyễn

Sửa bản in: Minh Như

Bìa: Hữu Nghĩa

In 4.000 cuốn, có kèm CD bài tập, khổ 16x24cm tại

Xí nghiệp In Machinco

21 Bùi Thị Xuân, Q.1, TP.Hồ Chí Minh

Số đăng ký kế hoạch xuất bản:

06-2006/CXB/102-1.10/PĐ

Cục xuất bản ký ngày 14 tháng 10 năm 2005

In xong và nộp lưu chiểu tháng 12 năm 2006

Sử dụng Crystal Reports XI

- Học Crystal Reports từ kinh nghiệm của chuyên viên thiết kế, cách tạo báo biểu trong ứng dụng .NET và Java.
- Cung cấp một cách rõ ràng, dễ hiểu kinh nghiệm và kiến thức thiết kế ứng dụng liên quan đến báo biểu.
- Trình bày các chức năng cấp cao của Crystal Reports mà bạn cần như: Sub-Report, Cross-tab, Charts và biến tham số động...
- Giới thiệu các cách triển khai và tích hợp trên các nền ứng dụng Visual Studio 2005, Windows Forms, ASP.NET, JSP và BusinessObjects Enterprise.
- Kèm theo sách là một CD-ROM với tất cả các đoạn mã và các báo biểu ví dụ có trong sách.



Minh Khai



Giá: 69.000 đ