

# **Pro Crystal Enterprise/Business Objects XI Programming**



Carl Ganz, Jr.

## **Pro Crystal Enterprise/Business Objects XI Programming**

**Copyright © 2006 by Carl Ganz, Jr.**

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN-13: 978-1-59059-759-0

ISBN-10: 1-59059-759-1

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Lead Editor: Matthew Moodie

Technical Reviewer: Ryan Follmer

Editorial Board: Steve Anglin, Ewan Buckingham, Gary Cornell, Jason Gilmore, Jonathan Gennick, Jonathan Hassell, James Huddleston, Chris Mills, Matthew Moodie, Dominic Shakeshaft, Jim Sumser, Keir Thomas, Matt Wade

Project Manager: Sofia Marchant

Copy Edit Manager: Nicole LeClerc

Copy Editor: Ami Knox

Assistant Production Director: Kari Brooks-Copony

Production Editor: Ellie Fountain

Compositor/Artist: Kinetic Publishing Services, LLC

Proofreader: Linda Seifert

Indexer: Kevin Broccoli

Cover Designer: Kurt Krames

Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail [orders-ny@springer-sbm.com](mailto:orders-ny@springer-sbm.com), or visit <http://www.springeronline.com>.

For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710. Phone 510-549-5930, fax 510-549-5939, e-mail [info@apress.com](mailto:info@apress.com), or visit <http://www.apress.com>.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at <http://www.apress.com> in the Source Code/Download section. You will need to answer questions pertaining to this book in order to successfully download the code.

*To Wendy, my beloved wife and inspiration, without whose patience,  
support, effort, and friendship this book would not be possible.*

# Contents at a Glance

Foreword .....	xv
About the Author .....	xvii
About the Technical Reviewer .....	xix
Acknowledgments .....	xxi
Introduction .....	xxiii
■ <b>CHAPTER 1</b> Welcome to BusinessObjects XI .....	1
■ <b>CHAPTER 2</b> BusinessObjects XI Server Architecture .....	13
■ <b>CHAPTER 3</b> Administration Tools .....	47
■ <b>CHAPTER 4</b> Using the Central Management Console .....	73
■ <b>CHAPTER 5</b> BusinessObjects XI SDK Programming I .....	103
■ <b>CHAPTER 6</b> BusinessObjects XI SDK Programming II .....	149
■ <b>CHAPTER 7</b> Crystal Reports and BusinessObjects XI .....	199
■ <b>CHAPTER 8</b> Programming the Report Application Server .....	229
■ <b>CHAPTER 9</b> Enterprise Solutions Using the BusinessObjects XI SDK .....	281
■ <b>CHAPTER 10</b> Security .....	337
■ <b>CHAPTER 11</b> BusinessObjects Unified Web Services SDK .....	367
■ <b>CHAPTER 12</b> Third-Party Solutions .....	395
■ <b>INDEX</b> .....	437

# Contents

Foreword .....	xv
About the Author .....	xvii
About the Technical Reviewer .....	xix
Acknowledgments .....	xxi
Introduction .....	xxiii
<b>CHAPTER 1    Welcome to BusinessObjects XI .....</b>	<b>1</b>
History .....	1
From Report Writing to Business Intelligence .....	2
BusinessObjects XI Release 2 .....	3
Licensing .....	4
Service-Oriented Architecture .....	5
Reporting Considerations .....	6
Preprinted Forms .....	7
High-Volume Printing .....	7
Legal Issues .....	8
Availability and Distribution .....	8
BusinessObjects XI vs. SQL Server Reporting Services .....	9
Market Considerations .....	9
Feature Differences .....	11
Summary .....	12
<b>CHAPTER 2    BusinessObjects XI Server Architecture .....</b>	<b>13</b>
InfoStore .....	13
FileStore .....	14
Servers .....	16
Programmatic Access .....	19
Central Management Server .....	22
File Repository Servers .....	24
Event Server .....	26
Page Server .....	29
Job Servers .....	32

Cache Server .....	34
Report Application Server .....	37
Server Groups .....	39
Creating a Server Group .....	41
Adding Servers to a Server Group .....	41
Extracting the Servers in a Group .....	42
The Auditing Database .....	42
Scalability .....	43
Backups and Disaster Recovery .....	44
Summary .....	45

### ■ CHAPTER 3    **Administration Tools** ..... 47

Central Configuration Manager .....	47
Properties Tab .....	48
Dependency Tab .....	49
Connection Tab .....	50
Configuration Tab .....	50
Protocol Tab .....	53
Import Wizard .....	53
Publishing Wizard .....	57
Administration Launchpad .....	61
Central Management Console .....	61
InfoView .....	62
Administration Tool Console .....	62
Business Views Manager .....	68
Crystal Reports Explorer .....	69
Summary .....	72

### ■ CHAPTER 4    **Using the Central Management Console** ..... 73

Folders and Reports .....	73
Folders .....	74
Reports .....	75
Objects .....	96
Calendars .....	97
Events .....	98
File and Schedule Events .....	99
Custom Events .....	101
Summary .....	102

<b>CHAPTER 5</b>	<b>BusinessObjects XI SDK Programming I</b>	103
	Connecting to BusinessObjects XI	103
	Connecting with User IDs	104
	Connecting with Tokens	106
	Connecting with Trusted Connections	108
	Examining the Connection Objects	109
	Querying the InfoStore	112
	The InfoStore SQL Language	113
	Working with the InfoStore	125
	Extracting Folder Information	126
	Creating Folders	129
	Deleting Folders	130
	Report Refresh Options	131
	Scheduling Reports	131
	Report History in the CMC	132
	Building Your Own Report History	133
	Adding New Reports	138
	Run Now	139
	Retrieving and Saving Schedules	140
	Adding Executable Code	144
	Handling Report Parameters Generically	145
	Summary	148
 <b>CHAPTER 6</b>	 <b>BusinessObjects XI SDK Programming II</b>	 149
	Scheduling	149
	Notifications	149
	Alerts	152
	Report Format Options	156
	TextFormatTabSeparated	159
	TextFormatPaginated	159
	TextFormatCharacterSeparated	160
	RichTextFormat	162
	RichTextEditableFormat	163
	PlainTextFormat	163
	PDFFormat	163
	ExcelFormat	164
	ExcelDataOnlyFormat	165
	Destination Options	166
	Printer Options	170

Programming Categories .....	171
Programming Events .....	173
Programming Calendars .....	175
Data Access Techniques .....	177
BO XI Web Controls .....	179
Getting Started .....	179
Web Controls .....	187
.NET Providers .....	192
Membership Provider .....	193
Using the .NET Providers .....	194
Summary .....	197

## ■ CHAPTER 7    **Crystal Reports and BusinessObjects XI** ..... 199

Crystal Reports Viewer Control .....	199
Setting Up the Control .....	199
Communicating with the BusinessObjects XI Server .....	204
Running a Report on Disk .....	205
Passing Parameters .....	207
Passing an Entire SQL Statement to Crystal Reports .....	208
Embedded vs. Nonembedded Reports .....	209
Exporting Reports .....	213
Filtering .....	215
Field Structure .....	216
Sections .....	217
Database Connectivity .....	220
Printer Options .....	220
Summary Information .....	220
Unmanaged RAS .....	221
Passing Data Sources .....	222
Crystal Reports Web Services .....	224
Summary .....	227

## ■ CHAPTER 8    **Programming the Report Application Server** ..... 229

Introduction to RAS Programming .....	229
Working with Reports .....	230
Connecting to the Data Source .....	231
Building the Body of the Report .....	236
Exporting Reports .....	265
Setting Parameters .....	267



Filtering Reports .....	268
Sorting .....	269
Charts .....	272
Report Options .....	274
Summary Information .....	276
Printing Reports .....	277
Saving Reports .....	279
Summary .....	280

## ■ CHAPTER 9 Enterprise Solutions Using the BusinessObjects XI SDK .....

On-Demand Web Service .....	281
On-Demand Web Service Internals .....	282
Scheduling Assemblies .....	289
Creating Criteria Screens .....	294
Building the WinForms Interface .....	295
Web Interface .....	308
BO XI Windows Service Monitor .....	313
Reporting Against the InfoStore .....	318
GetReportTree Web Service .....	319
Reporting Against the XML Output .....	325
Building Dynamic Menus .....	327
Storing Custom Metadata .....	330
Using Your RDBMS .....	330
Custom Properties .....	331
The Limits of Crystal .....	334
Summary .....	335

## ■ CHAPTER 10 Security .....

Security Considerations .....	337
Managing Security Through the CMC .....	338
Creating Users .....	339
Creating Groups .....	341
Adding Users to Groups .....	342
Extracting the Users in a Group .....	343
Subgroups .....	345
Access Levels and Security Rights .....	347
Roles .....	348
Rights .....	349

Limits .....	355
Non-Enterprise Security Management .....	359
Validating NT Group Users .....	360
License Keys .....	362
Summary .....	366
 <b>CHAPTER 11 BusinessObjects Unified Web Services SDK .....</b>	 367
Configuring the Unified Web Services SDK .....	367
Why Use the Web Services SDK? .....	370
Programming Web Services .....	370
Services .....	370
Creating a Connection .....	371
Queries .....	373
Extracting Data .....	376
Working with Reports .....	380
Working with Folders .....	389
Managing Servers .....	390
Managing Security .....	392
Summary .....	393
 <b>CHAPTER 12 Third-Party Solutions .....</b>	 395
APOS Systems Inc. ....	395
AddressBook Gateway .....	395
Archive Manager .....	396
Bursting Manager .....	398
InfoScheduler .....	400
Instance Manager .....	402
Instance Monitor .....	404
Key Performance Indicator .....	404
Object Manager .....	406
RealTime Monitor .....	408
Report Package Booster .....	412
RunTime Manager .....	413
Solutions Kit for ESRI GIS .....	413
View Time Security .....	416
Software Forces .....	418
.rpt Inspector Enterprise Suite .....	418

Teleran Technologies.....	421
iGuard .....	421
iSight .....	422
CRD .....	424
Excel Solutions.....	425
Syncfusion's Essential XlsIO.....	427
SoftArtisans's OfficeWriter .....	431
Word Solutions.....	434
Summary .....	435

■ INDEX .....	437
---------------	-----

# Foreword

**D**ata. More data. Still more data.

Think application complexity and Internet systems are driving the torrential increase in data volume? Think again. Life's only going to get more interesting with Service-Oriented Architectures (SOAs), which will expose richer data sources less siloed than ever before. This means that the limitations related to what vendors control and choose to expose about an application's inner workings will become less restrictive over time (whether by design or as a side effect of the SOA movement)—leading to easier data access for a wide variety of systems.

Now, more than ever, tools make IT possible.

The previous statements are not nearly as daunting as they might seem. Products like Crystal only continue to get easier (depth of designers), richer (presentation delivery mechanisms), and more complete (flexible data-access layers and programmatic object models). Indeed, Crystal's very reason for existing is to simplify and enable the formatting of complex data into more usable forms—from simple banded reports to sophisticated drillable context-sensitive dashboards and exception-based push reports that do more than present a situation—it enables interaction with them. The line between operational and analytic reporting continues to blur—and the application end user only stands to benefit, and by extension, the savvy developer who understands reporting as more than a thankless boring task stands to win also.

But maintaining the appropriate depth of understanding to utilize these advanced reporting toolsets to their best advantage, requiring knowledge of SDKs, APIs, and, in their most evolved state, object models, is more important than ever. Virtually all products now incorporate object models to influence the behavior of all aspects of report development, deployment, security, and usage analysis. This is more than a trend: it's now a requirement to compete for virtually all reporting tools.

While the GUI designers embedded in products such as Crystal continue to evolve, it's inevitable that you'll need to exploit the inner workings of a toolset to create custom behaviors—the alternative is to build them yourself. Building your own wrappers seems like a good idea at first, until you begin to understand the breadth and depth already included with the product. Why build a narrow situational façade when it (probably) already exists, and ships with the product anyway? A great deal more value is generated by using what's already there—whether you define value as speed of development, flexibility of evolution and use, or replicable standardized deployment.

Or, put another way, this stuff can make you look pretty darn smart.

Crystal's SDK has a remarkable amount of depth, but it has been hard to make sense of the various layers in the object model—this book makes this task simpler.

Understanding the division of power between Crystal's various layers is a critical IT skill: this book will make that easier, parsing the important from the inconsequential—and perhaps most importantly, making a broad subject accessible in targeted bite-size chunks.

Fred Seyffert  
*President, VeriPoint LLC*

# About the Author

■ **CARL GANZ, JR.**, is president of Seton Software Development, Inc., a provider of software design and development services located in Raritan, New Jersey. He has an MBA in finance from Seton Hall University and is the author of three other books on software development as well as dozens of articles on Visual Basic, C#, and Microsoft .NET technology. Carl has created numerous solutions over the years using the Crystal Enterprise and BusinessObjects XI .NET and RAS SDKs. He is the president and founder of the New Jersey Visual Basic User Group and has been a featured speaker at software development conferences in both the U.S. and Germany. Carl and his wife, Wendy, live in Raritan, New Jersey, with their son, Carl III, their dog, Elke, and their cats, Jack and Jake. Contact Carl at [seton.software@verizon.net](mailto:seton.software@verizon.net).

# About the Technical Reviewer

■ **RYAN FOLLMER** is a technical lead for Ciber, Inc., an international system integration consultancy. He specializes in user interface development using the Microsoft .NET Framework. As a consultant for nearly 10 years, Ryan has developed multiplatform applications for the financial, life science, and service industry markets. Ryan lives in Pittsburgh, Pennsylvania, with his dog, Toby, and can be reached at [ryanfollmer@gmail.com](mailto:ryanfollmer@gmail.com).

# Acknowledgments

**T**here are several people whom I would like to thank for making this book possible.

Ryan Follmer, who performed a thorough technical review of the entire book to make sure that everything within is complete and accurate. His work was outstanding, and he caught some truly subtle issues. After working in varying capacities on about a half-dozen book projects over the years, I can honestly say that Ryan has been the best technical reviewer I've ever had the pleasure to work with.

The technical support staff at Business Objects, especially everyone at Elite Team B and developer support, who patiently answered so many of my rather esoteric questions.

The management and editorial staff at Apress—specifically Ewan Buckingham, Sofia Marchant, Matt Moodie, and Ami Knox—for their professional guidance and overall kindness in steering this project through to completion.

My clients, whose many challenges over the years have provided the foundation of literally all that is contained within these pages.

My wife, Wendy, and son, Carl III, for providing me with the love, affection, and support that makes all these efforts worthwhile.

Most importantly, thanks be to God for the ability to do this kind of intellectually demanding work.

# Introduction

**B**usinessObjects XI is a powerful middleware server product that allows you to distribute your Crystal Reports and BusinessObjects reports to the enterprise. Out of the box it offers an intuitive, feature-rich front end that allows system administrators to load reports to the repository as well as schedule them to run, track report histories, send notifications, monitor events, and many other features as well. It also offers a powerful object model that allows you to create custom solutions for your enterprise applications. It is this object model that is the topic of this book.

Though the interface is quite intuitive, the object model definitely is not. While the documentation and sample code have improved dramatically since the first release, it's still not at the quality level worthy of a product in this price range. My hope is that this book fills in that gap. BO XI does not offer the solutions out of the box that SQL Server Reporting Services offers. Currently, it's the Catch-22 of the reporting world. BusinessObjects offers a slick, mature interface with a clumsy server object model (with no canned service-oriented programming solutions) and a poorly documented programming interface. SQL Server Reporting Services offers an ever-improving-but-still-not-close-to-a-Crystal-Reports-replacement interface with a well-documented and easy-to-use object model with server solutions provided right out of the box.

Crystal Reports was first released in 1991 and has been bundled with Visual Basic starting with the 3.0 release in 1993. This relationship has continued through the 2.0 release of Visual Studio .NET in 2006. Of course, since Microsoft released SQL Server Reporting Services in 2002, where the Microsoft/Crystal relationship is ultimately going still remains to be seen. In 2002, Crystal Decisions released the first version of its enterprise server known as Crystal Enterprise 8. This was really version 1 of the enterprise product as the number 8 was used to keep the version numbering in sync with the incarnation of Crystal Reports on the market at the time. In 2003, Business Objects purchased Crystal Decisions and began the rebranding effort toward the BusinessObjects name. The current incarnation of the server product, BusinessObjects XI Release 2, is a synthesis of server technologies that supports both Crystal Reports XI and Business Objects 6.5. The Crystal Reports product has even been enhanced to access BusinessObjects universes.

Crystal's technology is superior to that of the BusinessObjects product, which is one of the reasons Business Objects bought the company. The good news for Crystal developers is that rather than "BOizing" the Crystal product, Business Objects is "Crystalizing" their original product. Therefore the structure of the Crystal Enterprise object model will continue to be expanded to support BusinessObjects reports and will be with us for many years to come.

I started using Crystal Enterprise with the 9.0 version, and at this writing have been working with the product for over three years. Each release has offered major improvements over the last one without revolutionizing the object model to such a degree that a code rewrite was needed. The BusinessObjects XI release is no exception. The goal of this book is to explain the product from a developer's perspective. The BusinessObjects XI Management Console is an administrator's tool that is not intended to be offered to every user of the system, and the InfoView end user



interface may not suit all your needs. Therefore, I'll explain how to build solutions to handle on-demand reporting, scheduling, notifications, server management, report histories, security and user management, and many other tasks. In essence, you'll learn how to build all the features offered by the Central Management Console and InfoView. I'll accomplish this by examining production code adapted from real-world applications that are in use at several of my clients. We'll look at the thinking process behind the architectural decisions for the code samples as well as attempts that failed as I learned by trial and error. My hope is that you can use this code, with little or no modification, in your own production applications.

This book focuses mainly on the programming aspects of BusinessObjects XI, approaching the product from the perspective of a Crystal developer. Therefore, BusinessObjects-specific technologies such as Web Intelligence, Desktop Intelligence, and universes are not covered. Chapter 1 introduces BusinessObjects XI, reviews licensing issues, and compares its features with SQL Server Reporting Services. Chapter 2 discusses the server technology. Chapter 3 explains the tools that ship with the product, such as Import Wizard and Publishing Wizard. Chapter 4 covers the entire Central Management Console from a user's point of view. Chapters 5 and 6 introduce the object model of the .NET SDK. If you're already familiar with BusinessObjects XI as a system administrator, you may wish to stop here first. Chapters 7 and 8 cover the programming extensibility of Crystal Reports and the Report Application Server (RAS), respectively. Chapter 9 explains how to develop enterprise solutions using BO XI and is illustrated with a number of real-world production applications. Chapter 10 covers security, and Chapter 11 explains the new Unified Web Services SDK. Finally, Chapter 12 discusses some of the third-party products available for BusinessObjects XI.

Carl Ganz, Jr.  
*Seton Software Development Inc.*  
*Raritan, New Jersey*  
seton.software@verizon.net



# Welcome to BusinessObjects XI

**B***usinessObjects Enterprise XI* is a server-based, middleware product that allows you to distribute your Crystal and BusinessObjects reports throughout your organization. In addition to reports, you can also register compiled EXEs that can run customized business logic. You can upload static documents—Excel, Acrobat, Word, and text files, and so on—so as to publish them to the enterprise. BO XI is a scalable tool that can be deployed across multiple servers so as to handle high-volume report access. With its FTP functionality, you can even use BO XI as a conduit to other computer systems. It offers scheduling, report history, notifications, events, a security model, server management, and much more. Most of the features offered by BusinessObjects XI are exposed through APIs contained in a set of SDKs. Regardless of how you use BO XI, you'll likely take advantage of this object model to develop some kind of a customized front end. These object models and how to develop enterprise solutions with them is the subject of this book.

## History

*Crystal Reports* was first released in 1991 and achieved market share quickly, rapidly overtaking R&R Report Writer as the dominant report-writing tool on the market. In 1993, Crystal was bundled in Visual Basic 3.0, the first version of Microsoft's flagship development language that had any real database capabilities. Its inclusion in what became the most popular Windows development language ever launched Crystal Reports into its position as a de facto standard.

As good a product as Crystal Reports is, for the first decade of its existence it didn't have a server product behind it as did other competing business intelligence tools such as Cognos, Actuate, or BusinessObjects. To fill this gap, Crystal Decisions in 2002 released its *Crystal Enterprise* product, which allowed you to make reports available from a server with a zero-client footprint. Anyone who has ever developed applications across multiple pre-.NET versions of Crystal Reports knows that installing an application using one version of the Crystal runtime on the same machine as an application using another version of the Crystal runtime routinely resulted in unpleasant DLL conflicts. In many cases they simply wouldn't work together, as Crystal's technical support would often reluctantly admit.

In 2003 one of my clients paid \$150,000 for a two-processor license for Crystal Enterprise 9. They had been using Crystal Reports since it was first released and had legacy applications going back to VB 3.0 and every version of Crystal shipped subsequent to it scattered throughout the enterprise, each using different versions of the runtime. This client maintains a matrix of which applications use which versions of Crystal Reports and therefore would or would not

be compatible on the same machine. All these troubles disappeared when we brought Crystal Enterprise and its zero-client footprint online. The elimination of the headaches caused by the DLL conflicts alone was worth the price of the Crystal Enterprise 9 license.

Crystal Decisions was originally part of Seagate Technology, a manufacturer of disk drives. It was spun off in 2000 when Seagate was acquired and had been a privately held concern ever since. The company was intending to go public in May 2003, and its IPO registration listed revenues of \$271 million and net income of \$27 million. That year, Business Objects purchased Crystal Decisions for \$820 million.

The technology offered by Crystal Decisions complements that of Business Objects very well. Crystal Reports is first and foremost a report-writing tool. Such a tool is used by developers and very sharp end users who understand their relational data models, understand SQL, and can pull together a report. Business Objects is first and foremost a business intelligence firm whose flagship tool reports from universes. A *universe* is a metadata layer created by a database professional that shields end users from the intricacies of their RDBMS. Therefore, it is much easier for a nontechnical end user to create a report from a universe than directly from an RDBMS.

With this acquisition, Business Objects plans to keep both reporting technologies, as they each fill very specific niche markets. The server product, BusinessObjects XI Enterprise, will support distribution of both sets of reports—Crystal and the Web Intelligence and Desktop Intelligence reports offered by BO—to the enterprise. Moreover, both products will begin to share some of their core technology with one another. Not only will BusinessObjects XI host traditional BO reports, Crystal reports will be able to access data in universes.

In a July 2004 interview with *Computerworld*, Bernard Liautaud, the current chairman and then CEO of Business Objects, said

*Crystal will use the semantic layer of Business Objects and use the Business Objects [data] universe. Business Objects has built its success in query technology at the semantic layer and Crystal Decisions has built it on reporting. . . . Now the two things will be together in one product. We've got a unified portal and dashboard for both Crystal Decisions and the Business Objects products for the user to log in with a single user name and password and see all Crystal reports or do a document search and link reports. It's a true common user experience between the two products.*

In November 2005, BusinessObjects XI Enterprise was released, and the product is currently in its second release, known as R2. The tool allows both Crystal developers and BusinessObjects users to approach the server tool from their own technology base.

## From Report Writing to Business Intelligence

The terms *report writing* and *business intelligence* are often used interchangeably. Though both terms are related, they are not synonyms. Crystal Reports is primarily a report-writing tool. Its purpose is to allow developers to build reports against RDBMSs and other data sources and output their contents to the user. Though it can be used by nonprogrammers, it is aimed at the developer market. Crystal is known as a *banded report writer*. This means that a report

operates on a band hierarchy where one band can represent subtotal information, another band holds the total level, and another band displays the grand total level. Other bands represent page and report headers and footers. At runtime, the placeholders in each of these bands are replaced with the data they are linked to in a data source to produce a report. The basic hierarchy of a banded report writer is shown in Figure 1-1.

Report Header
Page Header
Details
Page Footer
Report Footer

**Figure 1-1.** *Banded report writer*

Over the years, Crystal's reporting tools have evolved with the addition of Crystal Analysis. *Crystal Analysis* is an On Line Analytical Processing (OLAP) tool that allows you to connect to the dimensions and cubes of your OLAP back ends to perform analysis and drilling into the data.

*Business intelligence*, commonly known by its initials BI, is an industry term that refers collectively to the processes and technologies involved in the collection and analysis of business information so that the organization can use it to make informed decisions. Often, BI systems are referred to as *decision support systems* or to a lesser degree, *executive support systems*.

One of the principal buzzwords in the BI world is *key performance indicators*, also known as KPIs. KPIs allow an organization to measure and monitor, in as real time as possible, the performance of the enterprise. KPIs will vary across industries, but they may include such metrics as week-by-week sales trends, current headcount, percent of sales goals achieved to date, or number of phone calls to technical support. These are the types of summary numbers that would likely appear on an executive dashboard.

As you can see, BI is far more comprehensive than simple report writing, if only because nontechnical business professionals are intended to accomplish it. These professionals are then free to focus on the metadata that abstracts their information rather than the technical details of using a complex report-writing tool and navigating often labyrinthine database structures while optimizing SQL statements.

## BusinessObjects XI Release 2

The key word in the name BusinessObjects Enterprise XI is *Enterprise*, and as this word implies in any product name, it is not cheap. Since the product is intended as a centralized reporting middleware solution that services the entire organization, large companies can focus on assigning software professionals skilled in the tool and its SDKs to service the product full time and so eliminate the need for every department and workgroup from developing and maintaining their own solutions.

## Licensing

Licenses can be acquired either by named user or by processor. Named user licenses cost about \$2,000 per seat. I have a client who prefers the user approach because the client has multiple departments that use BO XI, and each department has the product installed on its own servers. Since the load on any given server isn't sufficient to justify an entire server license, user licenses make more sense for that client.

Business Objects prefers not to publicize the cost of its enterprise server licenses. Each customer configuration is unique, and there are a number of variables that come into play when computing a license cost. Here is the company's official policy:

*BusinessObjects Enterprise is available in two editions, Professional and Premium with two flexible licensing options: Named User and Processor-based. For specific pricing information, please contact Business Objects, or a certified Business Objects partner directly.*

I do know from past purchasing experience that a two-processor license of BO XI costs in the neighborhood of \$150,000. The nonproduction versions of these licenses are usually offered at 50 percent of the full price. Note also that multicore processors require licenses for each core. There are a number of technical support options as well, and you can find the details on them here: [http://support.businessobjects.com/programs/enterprise\\_bi\\_products.asp](http://support.businessobjects.com/programs/enterprise_bi_products.asp).

Because BusinessObjects Enterprise XI is a rather expensive product, it may not be financially viable for all organizations. As its name states, it is enterprise software, and even a large organization may not even need something that may well only be used by a single department or workgroup. Fortunately, there is a version of the tool, called *Crystal Reports Server*, that is scaled and priced for smaller organizations.

The base Crystal Reports Server product comes with a five-concurrent-access license. With a license cost of \$7,500, most small to medium-sized organizations can afford it. You can scale the number of concurrent users up to a maximum of 20 concurrent access licenses, with each block of 5 licenses up to this limit costing an additional \$7,500 for a maximum cost of \$30,000. When one more than the maximum number of users attempts to log on, they will be denied access until an existing user session either logs off or expires.

You can only use Crystal Reports Server to run Crystal reports, and it cannot be scaled across multiple servers. The single Windows or Linux server you can run it on is restricted to a maximum of four processors. There is no Unix edition. Other than these restrictions, Crystal Reports Server has most of the same features as the Enterprise version.

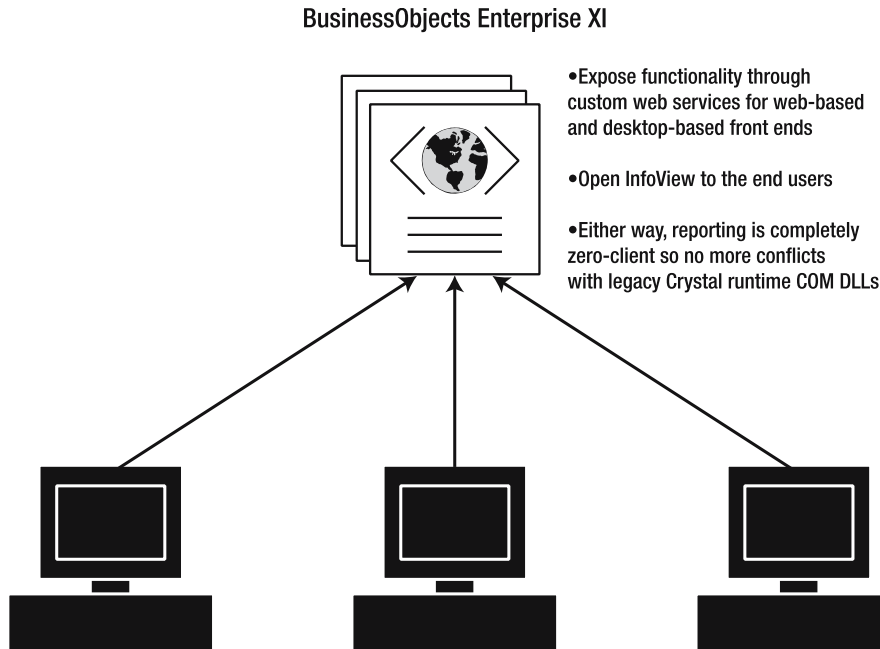
Business Objects changed the licensing model once it purchased Crystal Decisions, which, starting with Crystal Reports 8, allowed much more flexibility than was available before. Previously, the licensing model prevented you from installing Crystal Reports on a machine and building software that instantiated its objects so as to act as a report server or, in effect, a replacement for Crystal Enterprise should you choose to add enough features. If you wanted to do this, you needed a broadcast license. Broadcast licenses were not cheap. A license for 500 users would cost you \$10,000 per year, 1,500 users would cost you \$25,000 per year, and more than 1,500 users would cost you \$50,000 per year (a user being any individual who during the course of the year utilizes the report server at least once). The main goal of Crystal Decisions' licensing structure was to prohibit you from making your 8.x+ reports available to the enterprise without purchasing the rather expensive BO XI Enterprise license. I've read various posts to Internet newsgroups that refer to this licensing scheme as "extortionate" and "gouging."

Business Objects has removed these restrictive licensing requirements. If you wish to do so, you may install Crystal Reports on a server machine and develop your own report server as a few third-party companies have. By storing report schedules and on-demand requests in a database, you can create a Windows service that continually queries the report server data tables and, based on the attributes, creates a thread that instantiates the Crystal Reports API to run a report. This change in licensing is what allowed the CRD product from ChristianSteven Software, Ltd. ([www.christiansteven.com](http://www.christiansteven.com)) to be significantly cheaper to deploy, as a broadcast license is no longer necessary. This tool replaces much of Business Objects XI by sitting on top of your Crystal Reports DLLs, as well as implementing its own unique functionality, and is discussed in greater detail in Chapter 12.

The Crystal Reports Developer Edition offers a license for unlimited deployment of the reporting engine components within the enterprise. There are no additional licensing fees involved should you wish to build an application that uses these components and make that application available to the enterprise. Should you develop a product for distribution to other organizations, you'll need to purchase a license for each organization that uses your tool.

## Service-Oriented Architecture

One of the leading buzzwords in the IT field today is *service-oriented architecture* (SOA). As a middleware application, BusinessObjects XI certainly lends itself to an SOA implementation. Given the rise of web services since the release of .NET in 2002, the SOA approach has taken off and gained acceptance in the industry. BO XI is essentially a service, specifically one that provides reporting services to the enterprise. This functionality is exposed through the user interface of its management console and through its object model SDKs. By creating a web service wrapper to this object model, you can expose BO XI's power to the enterprise as illustrated in Figure 1-2. Creating web service wrappers for the BO XI SDK is explained in greater detail in Chapter 5.



**Figure 1-2.** *SOA architecture for BusinessObjects XI*

SOA is a service-based approach to application interoperability. It shares data using ASCII-based XML that is completely platform independent, enabling a common way for components to communicate with each other, and in doing so expose their functionality to the larger service ecosystem. Using a web service paradigm, you get a self-descriptive component (WSDL), a common invocation mechanism (SOAP), and a common data-exchange format (XML), all communicating across common communication layers (HTTP, among others). SOA allows you to build reporting functions that aren't an architectural afterthought, but rather grounded in a reliable methodology for integrating reporting functionality with every application in the enterprise.

Business Objects recently released their Web Services SDK, which implements this service-oriented architecture. The web services object model differs in varying degrees to the one offered in the .NET SDK. Chapter 12 is dedicated to the Web Services SDK.

## Reporting Considerations

Before embarking on any reporting project, there are some practical considerations to keep in mind. A very broad definition of reporting would be “anything that comes out of a printer.” If this is your mindset, then there are a few issues to consider when handling certain types of reporting projects.

Report distribution has made huge leaps since the introduction of the Internet and corporate intranets. Since many reports are built in to applications (compiled EXEs), users would normally need a copy of that application installed on their desktops in order to access the report. Now, the reporting functionality can be completely Web-based. Using Web-based

technology, the comparatively finite number of desktop applications can still have access to report functionality via web services, while the often much larger number of web users can access a Web-based application that will provide them with report criteria screens allowing them to filter, run, and export reports.

## Preprinted Forms

Oftentimes you may need to print a report on a preprinted form, such as an IRS 1040 or a W-2. If this is the case, you should examine whether the application really needs to print specific pieces of data at specific locations on the preprinted page, or simply reprint the entire form itself on blank paper. I have a client who was once spending \$7,000 per year on preprinted forms. When I showed him that the same result could be achieved if he performed a mail merge with a document composed in Microsoft Word, he was naturally elated at the annual savings. Moreover, he now has the flexibility of being able to change the document at will without coordinating the forms printer with the software developer.

## High-Volume Printing

High-volume printing raises issues that must be examined for cost efficiency. It's very easy to create a mail-merge application; quite another thing to manage it hands on. One of my clients is an elected official who printed his constituent reports on an ink-jet printer. Ink-jet printers are relatively cheap, and a decent one can be purchased for about \$100. The hidden costs here are the ink cartridges. With cartridges costing about \$40 each, you don't want to do high-volume printing on an ink-jet printer. In fact, ink-jet printers are loss leaders for the printer industry, as the real profits are in the replacement ink cartridges. Since they are used up at a much faster rate than laser toner cartridges costing twice as much, you'll end up spending more on ink-jet replacements than you will on toner. There are companies that also sell recycled toner cartridges at a sizable discount compared to new cartridges. In addition to this, ink-jet printers operate at only a fraction of the speed of laser printers. When my client indicated he wanted to print an annual newsletter to send to his 5,000 constituents, I recommended a laser printer. He now prints the text of the newsletter and merges the addresses as well.

Mass mailing can be a science unto itself. If your client needs do customized high-volume mailing pieces—for example, a monthly telephone bill as opposed to a supermarket circular where every recipient receives the same one—you may wish to examine an automated mailing solution. I have a long distance reseller client who sends thousands of telephone bills every month. The billing software creates print images of the bills and sends them to a Pitney Bowes machine that prints the bill, folds it, places it into a postage-paid window envelope, and seals it. Machines such as those sold by Pitney Bowes are designed specifically for this type of high-volume mailing, and the staff are experts on how to sort and organize the letters to achieve the maximum bulk-rate postage. If you have a requirement for customized high-volume mailing, examine the product solutions at [www.pb.com](http://www.pb.com).

If your volume of mail is so great that it cannot be realistically done in-house, consider the services of a bulk mailing company. These firms have the computers and printing equipment to print millions of documents and mail them within a few days. Mailing firms are large, high-volume facilities, some of which print and mail several hundred million documents per year. You can use your software to produce print images according to specifications given to you by the mailing company and then e-mail the data to their facility. Optionally, you could simply give them the raw data, and they will create the print routines for you. Depending on cost, you



can work out with them where your work will leave off and theirs will begin. Because it doesn't matter where the mailing company is located, you can contract with virtually any in the country to obtain the most reasonable rates. Be open to the idea of a mass mailing company—you could very easily become overwhelmed with printing and mailing tasks in a high-volume environment.

## Legal Issues

One often-overlooked area of report design is the legal implications of system output. Usually, data that is intended for internal corporate use is not an issue. Output that is intended for use external to the organization can often bring with it significant legal liability. As a rule, every report, certificate, fax, statement, or letter that is intended for use outside the organization should be passed by legal counsel for approval. Seemingly innocuous documents may have enormous legal ramifications, and these ramifications may differ from industry to industry. For example, one of the first systems I ever built was a target market mailing system for the sales agents of a leading insurance company. A sales agent would buy a list of names, and my system would merge these names with a marketing letter. As I made the system more flexible, I allowed the users to create their own letters. They gladly did so without ever consulting with in-house legal counsel. There were no safeguards in place preventing an agent from sending out a letter promising, say, a guaranteed 20 percent return on a particular product. In the past, courts have ruled that such statements are contractually binding on the company who issued them.

Documents as simple as account statements or even invoices can be potential traps for legal liability. Alert counsel to the format of these documents and exactly what information they contain. Counsel may create a disclaimer that accompanies the text of the document as a safeguard. Even these disclaimers have potential pitfalls as some states do not recognize this proverbial “fine print” if the print is, literally, too fine. This means by law that if the font is too small, the courts will disregard the existence of the text as having been too small for someone to reasonably read.

Copyright is also a property right that must be guarded. The summarization of even publicly available data has long been held to be copyrightable. Therefore, the database of a company like Dun and Bradstreet, which compiles credit information on millions of business entities worldwide, has an intellectual property interest in the data it has collected and in the unique way it presents it. Therefore it is necessary for any reports that may present this data to individuals outside the organization to affix a copyright notice declaring the information in the report to be proprietary. Absence of a copyright notice can make prosecuting a case for infringement difficult to impossible, as it is incumbent on the copyright owner to make a clear and visible declaration of ownership.

A good rule of thumb is to contact corporate counsel before distributing any information to the outside. The attorneys (and your client) will be impressed by your foresight on these legal issues and will appreciate the fact that you consulted them.

## Availability and Distribution

The final design issue is to determine who needs to run the reports. Traditionally there is a Report option on the main menu of desktop applications from which criteria screens are launched and reports printed, displayed, and exported. While there may be only 20 users of your application, there may be 100 users of the reports. This is where hybrid application development comes into

play. Therefore, it may make sense to select some or all of the reports that are needed by users who are not users of the application itself and create them as web reports.

The Web is no longer the only way to make reports available to the masses. As PDAs (Personal Data Assistants) become more prevalent, more users will want to access their data remotely via a wireless connection to the server. Due to the current state of PDA and wireless technology, there are a number of limitations you'll need to deal with when allowing reports to be transmitted to a PDA. Because of the slow connection speed, you'll need to limit the amount of data you can transmit between the server and the PDA. Note that this speed problem occurs with wireless modems. People using PDAs with wireless LAN cards enjoy speeds above 10 mbps. This will affect the way you design your reports, especially the HTML exports that the PDA device often uses. Another problem is how to handle the ergonomics of a small screen that may or may not have a color display.

In many instances, users are more interested in receiving reports on a regular basis than they are in running them individually. This is known as the *"push" versus "pull"* approach. BO XI offers this in its scheduling feature. You can set a report to run at a certain time and with a certain frequency and then deliver the results in a desired format (Excel, PDF, etc.) to a printer/fax, e-mail address, or FTP server—the *"push"* method. Or, you can run reports on demand and download them to the client—the *"pull"* method.

## BusinessObjects XI vs. SQL Server Reporting Services

BusinessObjects/Crystal Enterprise and Crystal Reports have long competed with other report-writing and business intelligence solutions in the form of Actuate, Cognos, and Hyperion, among other competitors, and have always dominated market share. For years, Microsoft never entered the report-writing market, and with the exception of the reporting tools in MSAccess, never released a report writer. In 2002, that all changed when Microsoft released the first version of its SQL Server Reporting Services (SSRS). The second version was released three years later along with the premier of SQL Server 2005.

### Market Considerations

SSRS is Microsoft's first stab at a report-writing and business intelligence tool. Like BO XI, there are two main parts to the tool: a user interface to write reports and server software to schedule and distribute these reports. Currently, SSRS is offered as a *"free"* add-on to a SQL Server license. Since the recommended Microsoft configuration calls for a dedicated server to host SSRS, you need to purchase a SQL Server license for that server in order to obtain your free reporting software. Since SSRS and SQL Server are both resource-consuming applications, you don't want them battling one another for the memory and CPU of the same machine. Therefore, you'll want to install them on separate machines since it's not possible to allocate memory between the two products.

If you have a large centralized reporting application, you'll likely want to scale it across multiple servers to perform load balancing and implement redundancy. You need to purchase a SQL Server license for each of these servers as well in order to install your free copy of SSRS. Depending on your reporting needs, your free copy of SSRS could cost you up to \$100,000.

When Visual Studio 2005 was first announced, there was much concern among the developer community as to whether Microsoft would continue its relationship with Business Objects and bundle Crystal Reports with the .NET development suite. Microsoft promised to do so from the beginning and has kept that promise so far. Will Microsoft eventually drop the Crystal bundling and start offering only SSRS? Only time will tell. There may be some antitrust implications from doing this and offering only its own tool, as was the case in its battles with Netscape in the late 1990s. History, however, is not on the side of Business Objects. Lotus once dominated a spreadsheet market that is now controlled by Excel. WordPerfect once dominated a word processing market that is now owned by Word. dBase once reigned supreme in a desktop database market that it long ago surrendered to Access. Novell once overwhelmingly controlled the local area networking market but has been marginalized by Windows NT. Remember when IBM's OS/2 was once considered a "Windows killer"? I've read one respected Microsoft authority (nonemployee) who referred to SSRS as a "Crystal killer."

I'm not at all suggesting that BusinessObjects/Crystal Reports is dying and you should run to SSRS before it's too late. Quite the contrary, Business Objects is a thriving company and BO XI is a thriving product. With over 15 million registered copies of Crystal Reports throughout the world and a decade-and-a-half of developer experience, this tool isn't going anywhere anytime soon. SQL Server has been on the market for ten years now and has not managed to unseat or even marginalize Oracle. Reporting tools are one of the more difficult IT items to replace, though not quite as difficult as replacing a database. I'm merely stating that SSRS is now a player in the reporting market, it's not going away, and it needs to be carefully considered and respected as it grows to become a stronger rival. Watch this product closely and carefully.

Should you wish to move to SSRS from Crystal, the transition won't be an easy one. If you have a large base of Crystal reports, converting them to the Report Definition Language (RDL) format used by SSRS will not be a simple task. Hitachi Consulting ([www.hitachiconsulting.com/downloadPdf.cfm?ID=251](http://www.hitachiconsulting.com/downloadPdf.cfm?ID=251)) offers a tool call RDL Generator that will convert the majority of each report to RDL, and Jeff-Net ([www.rpttosql.com/faq.htm](http://www.rpttosql.com/faq.htm)) offers a service that does the same thing. Both companies can only perform a partial (60 percent or better) conversion. These products can convert page headers and footers, field position and formatting, special fields, formulas, groups, and parameters. Other features will not convert and need to be handled manually. The report formats are too different to allow the creation of a single tool that can take any Crystal report and output it to the RDL format without further modification required. It's not as simple as opening a Lotus spreadsheet in Excel.

At this writing (September 2006) the market has not overwhelmingly taken to SSRS. The best way to cut through the hype on any product is to see what companies are hiring for and are therefore willing to immediately spend money on. After checking with several IT recruiter contacts, they have all told me that Crystal Reports and CE/BO XI are still far more in demand than any other report development skill. SSRS appears to be still very much in the margin. A search of the Dice ([www.dice.com](http://www.dice.com)) and Monster ([www.monster.com](http://www.monster.com)) sites shows the number of job openings for Crystal Reports and Crystal Enterprise/BusinessObjects XI developers is dramatically larger than the demand for developers with SSRS expertise. The third-party book market has responded well to SSRS, as there have been as many titles published about it in the past 4 years as Crystal Reports has had published about it in 15 years, excluding multiple releases of the same title for different versions of the product.

## Feature Differences

SSRS reports use an XML format for the report definitions called RDL. Microsoft publishes the specification for RDL, as third-party developers are encouraged to develop technologies to complement SSRS. It had been hoped over the 4 years since the release of SSRS that someone would develop a user interface that would stand superior to the rather anemic one offered by Microsoft. So far, no one has stepped up to the plate. Crystal's interface has been refined over 15 years of use and 11 releases. SSRS's interface has been around for 4 years and 2 releases. Understandably, Crystal offers a far superior report designer interface to that of SSRS.

Crystal Reports and BusinessObjects XI both offer many features over SSRS. One of these advantages is the fact that the report-writing product and the server product are not married to one another. You need Crystal Reports to create reports, but you don't need the server middleware to run or distribute them. If you have an application that needs to be distributed to a large number of users, you can include the Crystal Viewer control and the necessary Crystal runtime DLLs with the installation set and you have a self-contained application. SSRS does not work this way. The reports can only run on the server middleware, and there are no runtime DLLs that will allow an RDL file to execute independent of the server software. Thus, SSRS would not work for applications distributed to the general public.

SSRS has fewer export options than BO XI—Excel, PDF, MHTML, TIFF, XML, and ASCII only. BO XI does not support TIFF, XML, or MHTML but offers many others that SSRS does not support. MHTML is a self-contained web file that embeds images within the report. It is the default rendering format for Internet Explorer 6.0. Both products allow you to create report schedules and distribution lists. SSRS calls these schedules *subscriptions*. You can create data-driven subscriptions whereby a list of users in an external data source are set to subscribe to a series of scheduled reports and can receive them via e-mail or disk file.

The concept of servers also exists in SSRS, only here they're called *processors*. When scheduling reports, for example, SSRS uses the SQL Server Agent service. Scheduled reports exist as jobs in SQL Server Agent. When a job executes, SQL Server Agent adds an event to the queue maintained in the report server database. The Scheduling and Delivery processor polls the queue periodically to determine what jobs are in the hopper. When the time comes to execute them, SSRS processes them accordingly.

Where BO XI has the Report Application Server (RAS) for programmatic report creation (see Chapter 8), SSRS offers a documented object model for the RDL format only. Each element of a report can be output to an RDL file via an `XmlWriter` object. This is a long way from the power of the RAS.

Another point to keep in mind when comparing the two products is that SSRS is primarily a developer's tool, not an end-user tool, though in SSRS 2005 Microsoft did introduce a designer that allows end users to create ad hoc reports. Crystal Reports is aimed at both developers and end users. There are Standard, Professional, and Developer editions to choose from, each with its own feature set tailored for the specific type of user.

One area where SSRS is superior to Crystal is in its documentation and Knowledgebase. Microsoft has traditionally offered excellent documentation, and SSRS is no exception. The Knowledgebase for SSRS also has a substantial amount of material that explains the different features of the product in more detail. Business Objects documentation is complete for the user interface of the product but rather weak for the .NET SDK, and even weaker for the RAS SDK. Its Knowledgebase does have some useful articles and technical papers but nowhere near the quality level of MSDN.

Though SQL Server Reporting Services is not nearly the mature product Crystal Reports/BusinessObjects XI is, it does offer one distinct advantage in the web services it exposes. BO XI offers a powerful, if not completely intuitive, object model. From this object model you can build enterprise solutions that provide all the reporting features you can imagine. The problem is that you need to build these solutions yourself.

SSRS on the other hand exposes a collection of web services that offer you access to the features of SSRS. Probably the most commonly used web method is the `Render()` method of the `ReportExecutionService` web service that allows you to run reports with specified parameters and return the output to the proxy application. Other web service methods allow you to read and set report parameters, retrieve report histories, set limits on the report history instances, and create schedules that determine which users will receive which reports when and in what format. Should you desire a custom interface to any of these features in BO XI, you'll need to create, deploy, and maintain your own web service application and write the code yourself.

## Summary

Business Objects can get you into the enterprise reporting world relatively cheaply with Crystal Reports Server. When your organization grows beyond the bounds of this tool, you have the unlimited resources of BusinessObjects XI awaiting you. Though Business Objects is firmly rooted in the reporting world, you still need to keep a careful eye on SQL Server Reporting Services as it gradually continues to gain market share.

Now that you know what BusinessObjects XI is, what it can do for you, and where it stands in relation to other reporting tools on the market, we'll start taking a look as to how it works. In the next chapter we'll discuss the collection of servers that together comprise BusinessObjects XI.



# BusinessObjects XI Server Architecture

**B**usinessObjects XI is a multitier, server-based product that comprises a number of logical servers. These servers run as Windows services, and they can all be installed on one machine or distributed across multiple machines, each running multiple processors as your needs demand. In this chapter, we'll examine how to manage those servers, command them programmatically, and extract information about them. We'll look at programming examples of how to work with servers and server groups. If you are unfamiliar with the basics of programming with the SDK, see Chapter 5 for an introduction.

## InfoStore

When you first install BusinessObjects XI, the product creates a series of tables in your designated database back end. This database is known as the *InfoStore*. The default RDBMS is SQL Server, but you can use almost any database on the market. The InfoStore tables hold the information about all your reports, folders, users, and object packages. In short, any data that you see in the Central Management Console (CMC)—except for the actual report and image files themselves—is stored in the InfoStore.

BO XI provides an InfoStore class in its SDK to act as an abstraction layer to these database tables. We'll examine this InfoStore class in much greater detail in Chapter 5 as it is the sine qua non of the entire object model. There is no reason for you to ever access the InfoStore RDBMS tables directly. Altering the database directly in any way will likely cause damage to your installation, and BusinessObjects can no longer be held accountable for technical support. The four RDBMS tables are `CMS_InfoObjects5`, `CMS_Aliases5`, `CMS_IDNumbers5`, and `CMS_Relations5`. The main table is `CMS_InfoObjects5`. This table stores all the data about your reports either indexed in named columns or unindexed in binary columns. A partial data display from this table is shown in Figure 2-1.