

Choosing the Right Business Objects SDK for Your Needs



Applies to:

BusinessObjects XI

Summary

Before you build your application, review your requirements to help you select the right SDK for your current and future needs. This document describes the eleven SDKs available in Business Objects products to help you choose the right one for your needs.

Author: Robert Horne, Product Manager - Embedded BI and Developer Strategy

Company: SAP

Created on: 16 October 2008

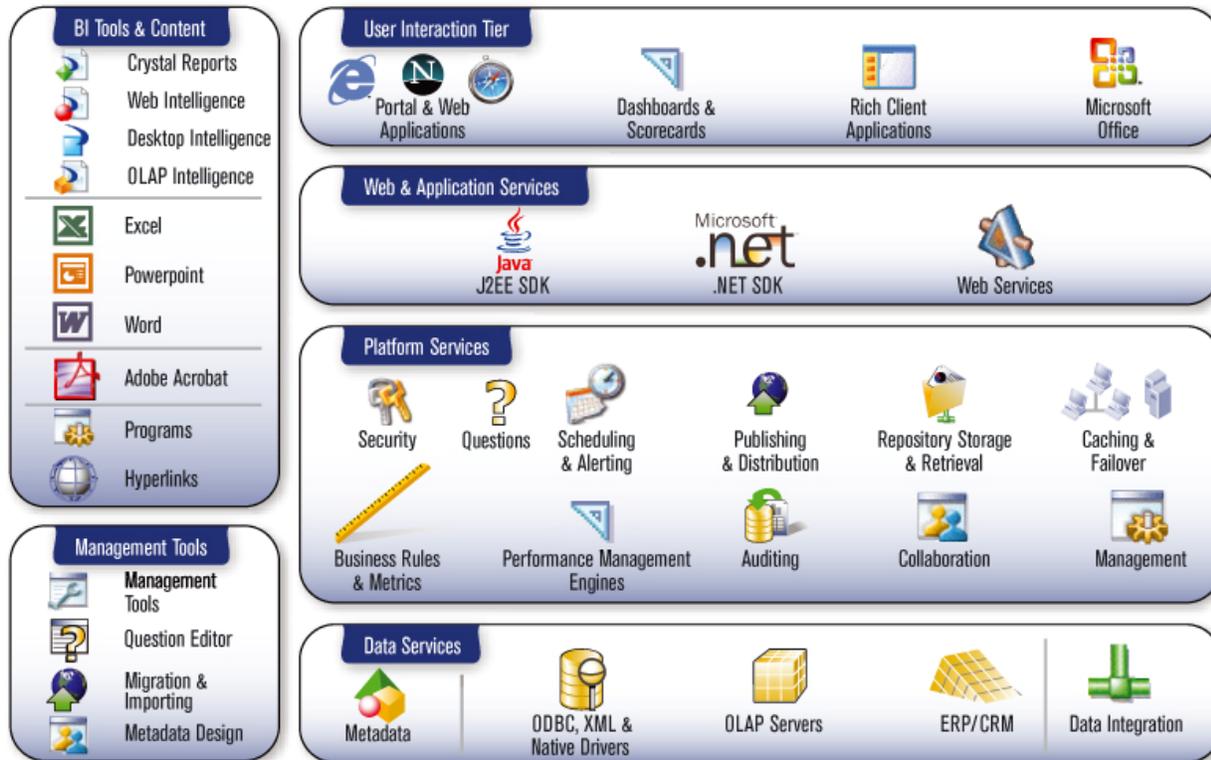
Table of Contents

Introduction	3
Business Objects provides a full stack of BI Products	3
What do you need to do today?	3
Business Objects API Deployment Scenarios.....	4
Component Deployment	4
Enterprise Server Deployment.....	4
Which SDK?	5
Enterprise SDKs.....	5
Web Services SDK.....	7
When to Use the Enterprise or Web Service SDK.....	7
Reminders.....	7
Component SDKS.....	8
Outgrowing the Component SDKs.....	9
Related Content.....	10
Copyright.....	11

Introduction

Often, several SDKs might appear appropriate candidates, but choosing the wrong one could waste a lot of your time. Discover which SDK "belongs" to each Business Objects product you've installed. Learn how to choose the best one and get your development project off to the right start. Understand the differences between the component SDKs and the Enterprise SDKs, when each is most appropriate, and how you can easily upgrade component technologies to server-based technologies. You'll leave this breakout with a solid understanding of how each SDK fits into the Business Objects and Crystal product roadmap, and how you can find resources to help your development projects.

Business Objects provides a full stack of BI Products



What do you need to do today?

- Do you need to create a desktop application or a web application?
- Do you need centralized report management?
- Do you have existing investments in Crystal Reports, Web Reports, Desktop Intelligence/Full Client Documents, or MS Office documents that needs to be integrated?
- What kind of load are you expecting on your application?
- How fast do your reports need to run?
- Architecture, are there numerous firewalls I need to navigate?
- Do I build security or do I want to piggy back off an existing security infrastructure?
- Do I need to modify or create reports at runtime?
- How will I scale this application as my needs grow?
- Do I have to use a specific programming language?

Business Objects API Deployment Scenarios

Business Objects has a number of APIs available to customize and embed reporting and query and analysis tools into your applications. Business Objects products provide two primary API deployment scenarios for embedding their products into third party applications: component deployment and server deployment.

Component Deployment

Component deployment is currently supported by our Crystal Reports product. In this deployment scenario, you can embed the Crystal Reports runtime engine inside third party application processes. This gives you the ability to embed reporting inside of your applications without having to build your own complex report engines. Business Objects currently provides these component APIs in both Java and .NET, and they are available in all of the leading developer IDEs. You can access these APIs through these products:

- Crystal Reports 2008
- Crystal Reports XI Developer
- Microsoft Visual Studio
- Crystal Reports for Eclipse
- Rational Application Developer

Enterprise Server Deployment

One of the limitations of the component deployment scenario is that the Crystal Reports engine is embedded in the same application process as the third party application. This means that they share the same application resources such as memory and CPU cycles. In some cases the load on the report engine can be very high given the number of end users and the popularity of the reports. In these scenarios we recommend moving to a server deployment where the report processing is moved to a separate server process.

Business Objects provides a number of products to support this deployment scenario. Our server products also provide features that allow for report scheduling, integrated LDAP and Active Directory security, load balancing, fault tolerance, and much more. Another benefit to using one of our server products is the addition to new reports types that support adhoc query and analysis and dynamic dashboard content. Like the component scenario the server products also have fully featured Java and .NET APIs. In contrast to the component APIs the server APIs are client APIs and control the functions of the server products. You can get access to these features through these products:

- Crystal Reports Server
- Business Objects Edge
- Business Objects Enterprise

Which SDK?

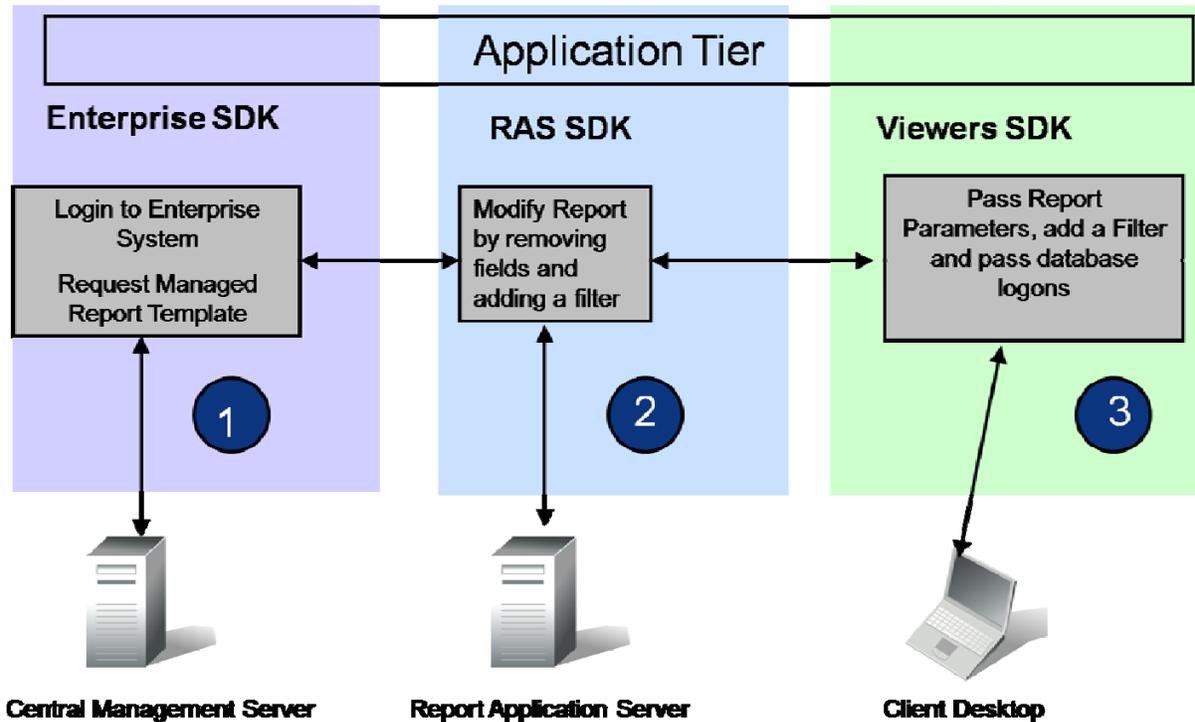
Enterprise SDKs

The BusinessObjects Enterprise SDK is comprised of all the following SDKs that work together to enable you to create applications with the functionality similar that found in Infoview and the Central Management Console (CMC):

- **Enterprise SDK** – With the Enterprise SDK, you can create Java, .NET, or ASP web portal applications, view all report types, use report creation and modification APIs, manage users, instances, groups, custom programs and events, and access security and scalability features.
- **Report Application Server (RAS) SDK** – Use when you want to create and modify Crystal reports at runtime, push data into reports with ResultSets, RecordSets, DataSets, POJOs, or Object Collections, and/or change your database at runtime (for example, from a development to production database). NOTE that if you do not need to modify reports at runtime, then it is recommended that you use the Page Server for optimal performance when viewing reports.
- **Universe Designer SDK**– Use this SDK to create and modify Universes using COM components.
- **Desktop Intelligence SDK** – Use this SDK to automate Desktop Intelligence documents. You can create VBA macros that encapsulate common commands and functions to run inside Desktop Intelligence applications, create plug-ins to add optional commands, features, and data sources, and implement custom events that run when runtime events are triggered.
- **Report Engine SDK** – The Report Engine SDK is available in .NET (RENET) and Java (REBEAN) and is used to view Desktop Intelligence and Web Intelligence documents. Use the Interactive Editor Integration sample for easier integration into custom viewing applications. For user authentication and security, use the Enterprise SDK.
- **Viewers SDK** – Use when you want to view Crystal reports instances or templates. The Viewers SDK is available in Java, .NET, and COM APIs and handles common viewing scenarios such as paging, exporting, searching, drilling-down, and printing, and common exporting scenarios such as PDF and Microsoft Office. This SDK also handles prompting and filtering – you can either allow the viewer to prompt users or you can set the values yourself in your application.
- Figure 1 provides an example of how three of the SDKs work together to modify and view a Crystal report.

Figure 1

Collaboration Example: Modify and View a Crystal Report



© 2008 SAP AG. All rights reserved.

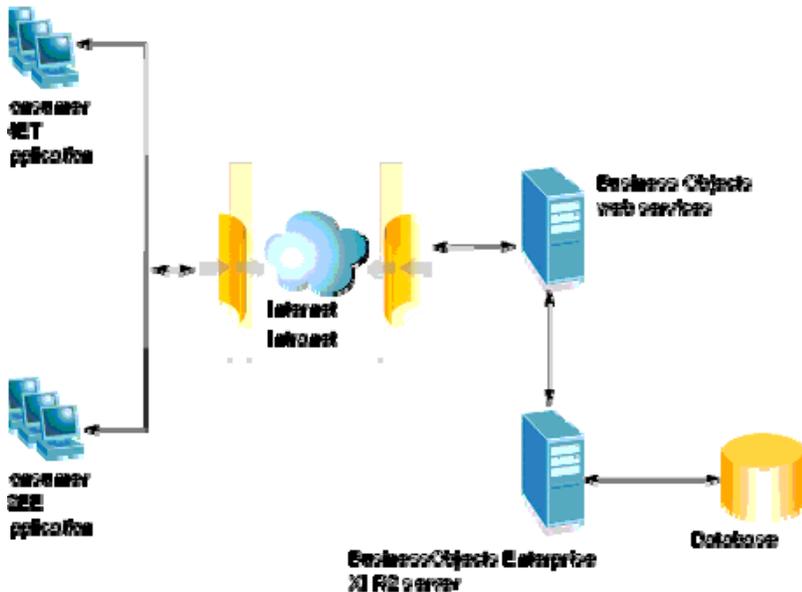
The BusinessObjects Enterprise SDK supports Java, .NET, and COM. Enterprise SDK applications can handle cluster failover, management of all objects such as reports, folders, users, programs, calendars, and events, security for all objects including users, groups, folders, and reports, and advanced scheduling of reports for non-peak hours, to different export formats, and using complex business rules using events.

Web Services SDK

The Web Services SDK supports .NET and Java client APIs and provides full access to all Enterprise plug-ins (for example, publishing and scheduling). This SDK allows you to view all report types, create smart clients that connect to enterprise data (for example, Xcelsius), easily connect through firewalls, create and modify queries using managed Universes, consume WSDL files directly to support other client languages, and share business intelligence services between companies and departments.

The loosely coupled application architecture of the Web Services SDK allows for easier integration. The SDK also provides simplified access to objects in the BusinessObjects Enterprise repository using a Universe Resource Identifier (URI) based query. Figure 1 shows a typical BusinessObjects Web Services Architecture.

Figure 1



When to Use the Enterprise or Web Service SDK

Use the Enterprise SDK when you need the highest performance possible, access to the Report Application Server (RAS) SDK to create or modify Crystal reports, and/or ability to create or modify Web Intelligence documents (reports). Use the Web Services SDK when you need to create smart client applications that connect to enterprise data, easy access from client to server through complex firewall infrastructures, and/or Web Service access to your enterprise server to fit some integration need. For more information about the differences between these two SDKs, download the white paper [Comparing the Enterprise and Web Services SDKs](#).

Reminders

Crystal Reports Server includes the same Enterprise SDKs as BusinessObjects Enterprise. However, Crystal Reports Server does not include or provides access to Web Intelligence, Desktop Intelligence, or universes.

If you do not need to create or modify reports at runtime, then do not use the Report Application Server (RAS) SDK to view reports. Instead, use the BusinessObjects Enterprise Page and Cache servers for optimal performance.

Component SDKS

Unlike the Enterprise SDKs, the Component SDKs do not work together and are available only in Crystal Reports:

- **Java Reporting Component (JRC)** for use in Crystal Reports for Eclipse, BEA WebLogic Workshop, IBM Rational Studio
- **Report Designer Component (RDC)** for use in Microsoft Visual Studio 6
- **Crystal Reports for .NET** for use in Microsoft Visual Studio .NET

Component SDKs Features

These SDKs provide customizable and embeddable viewers, royalty-free runtime for desktop application deployment, static report processing and viewing, complex runtime report creation and modification, and Web reporting components for your web site.

When to Use the Component SDK

It **IS** recommended to use one or more of the component SDKs when you

- Need a small self contained component desktop application
- Are deploying a small web application that will be accessed by a department or work group in your company
- Run most reports in an on-demand scenario
- Want to embed the reporting engine into your application process

When Not to Use the Component SDK

It is **NOT** recommended to use component SDKs under any of these scenarios:

- Mission critical web applications with report processing failover.
- Reports need to run at specific times, based on specific events, or on the successful completion of a third party business process.
- Share report objects like formulas, SQL commands, text objects and images in the object-oriented repository.
- Manage shared reports in company repository
- Web service access to enterprise data
- Complex and powerful semantic data layer required
- Robust security options for user, group, object, and folder levels
- Reliable batch report processing

Outgrowing the Component SDKs

When your needs outgrow the component SDKS, it is time to migrate to the Enterprise SDKs.

Viewer code

You can reuse the viewer code for Web based SDKs because all Java SDKs (JRC, RAS, and Enterprise) use the same Viewer SDK objects, all .NET SDKs (CR.NET, RAS and Enterprise) use the same Web Form Viewer, and the COM RAS and Enterprise SDKs use the same viewer objects.

Java Reporting Component (JRC) to Report Application Server (RAS)

The JRC uses the same code as RAS; therefore, to migrate to BusinessObjects Enterprise, change the import statements as follows:

	Java Reporting Component	Report Application Server
Import Statement	com.crystaldecisions.reports.sdk	com.crystaldecisions.sdk.occa.report.application
Report Object	Report = new ReportClientDocument("myReport.rpt");	report = oReportAppFactory.openDocument((IInfoObject) oInfoObjects.get(0),0, java.util.Locale.ENGLISH);

Crystal Reports .NET to Report Application Server

You can use your CR.NET code in your Enterprise application.

	Crystal Reports .NET	Report Application Server
Report Object	ReportDocument report = new CrystalDecisions.CrystalReports.Engine.R eportDocument(); report.Load("C:\myReport.rpt");	ReportDocument report = new CrystalDecisions.CrystalReports.Engine.ReportDocum ent(); report.Load("myReport in Enterprise", myEnterpriseSession); ...

Related Content

[Business Objects SDK Overview Diagram](#)

This object model diagram takes you through some key questions to help you decide which SDKs are most appropriate for your needs.

[Business Objects Community on the SAP Community Network](#)

Visit the Business Objects Community for additional developer resources such as the Developer Library, sample applications, and other white papers.

[Comparing the Enterprise and Web Services SDKs](#)

BusinessObjects Enterprise and Crystal Reports Server provide the Web Services and the BusinessObjects Enterprise SDKs. This document will help you decide which SDK to use to access Enterprise services.

Copyright

© 2008 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, System i, System i5, System p, System p5, System x, System z, System z9, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, POWER5+, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

These materials are provided "as is" without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

Any software coding and/or code lines/strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.