

# Crystal Reports

## Comparison of Crystal Reports development tools used in Microsoft Visual Basic

---

### Overview

This document provides an introduction and a comparison of the Crystal Reports development tools used in Microsoft Visual Basic. This document is for use with Crystal Reports 6 and 7. More detailed information on the various development tools can be found on our web site, in the Documentation section of our Technical Library at <http://support.crystaldecisions.com/docs>.

### Contents

<b>CRYSTAL ACTIVE X CONTROL (OCX) .....</b>	<b>2</b>
<b>CRYSTAL REPORT PRINT ENGINE API .....</b>	<b>3</b>
<b>CRYSTAL REPORTS AUTOMATION SERVER .....</b>	<b>3</b>
<b>REPORT DESIGN COMPONENT .....</b>	<b>4</b>
<b>CONTACTING CRYSTAL DECISIONS FOR TECHNICAL SUPPORT .....</b>	<b>5</b>

## Introduction

Crystal Reports has been an important tool for Visual Basic (VB) developers since it was first bundled with VB in 1993. Visual Basic developers commonly request a means of effectively incorporating their Crystal Reports into their Visual Basic applications. Crystal Reports provides the developer with several possible solutions. This document compares these solutions so users can determine which is most suitable for their requirements.

With Crystal Reports, Visual Basic developers can add sophisticated report generating and printing capabilities to their applications by using the Crystal Report Print Engine (Crpe.dll/Crpe32.dll). You receive royalty-free rights to ship the Print Engine with any application that you create. The complexity of your reporting is dependent on your reporting requirements and which reporting method you use to generate the reports.

Currently there are four development methods available within Crystal Reports to execute reports from a Visual Basic application:

- Crystal ActiveX Control (Crystl16.ocx/Crystl32.ocx)
- Crystal Report Print Engine API (Application Program Interface)
- Crystal Reports Automation Server
- Report Designer Component

## Crystal ActiveX Control (OCX)

The greatest advantage of the Crystal ActiveX Control (commonly referred to as the OCX) is its ease-of-use as there is a graphical interface. Use of the OCX is intuitive, as there is no need to write a series of direct function calls to display a report. The OCX allows the developer to quickly integrate Crystal Reports into an application without having to understand Crystal Print Engine API calls or the object hierarchy of the Automation Server.

The OCX is more limited in functionality than the other methods of executing reports. For example, in versions 5.0 and 6.0 of Crystal Reports, report designers are now incorporating subreports into their reports. Developers commonly request access to subreports from their applications. It is possible to access subreports using all four methods listed above, but not all the properties and methods that are available for the OCX control are available for use in the subreport. Currently, with Crystal Reports 6, only the most common properties/methods of the OCX are supported with subreports.

The OCX is the only method developers can use to bind a report to a data control or a Truegrid control. Since the OCX is a wrapper around the Print Engine, a report may take longer to load as opposed to using Print Engine API calls or the Automation Server.

If you have a version of Crystal Reports that came bundled with Visual Basic, the only development tool that is installed is the OCX control. In order to use

other development tools, you need to have the stand-alone version of Crystal Reports.

## Crystal Report Print Engine API

In many cases, using the Print Engine API calls provides more flexibility for both the developer and the end user. The Print Engine API calls contains more functionality than the OCX. The process for loading a DLL and calling DLL functions is a well-documented feature of the Windows API. Using Print Engine API method will also be the most code intensive development tool.

The Print Engine API calls allows the developer to GET information from an existing report and SET new information in the report. This functionality is very limited with the OCX. There are a few methods using the OCX that allow the developer to GET information from the report.

You can also access the Windows DEVMODE structure by using Print Engine API calls. Developers can then access information about a printer driver's initialization and configuration. Often, there is a need to change the printer orientation of a report (for example, from portrait to landscape) and this is only possible if you can access the DEVMODE. You cannot change the printer orientation of a report by using the OCX, as it does not allow you to access the DEVMODE structure.

The Print Engine API calls are only available in stand-alone versions of Crystal Reports.

## Crystal Reports Automation Server

The Crystal Report Automation Server was added as a development tool with the release of Crystal Reports 6. The Automation Server provides an object-oriented interface to the Print Engine to meet developers' advanced reporting needs.

The Crystal Reports Automation Server is an in-process automation server that is based on Microsoft's Component Object Model (COM) technology. This means that the Automation Server can be used in any development environment that supports COM. COM is a component software model that allows applications to communicate with one another. It does this, not by specifying how the components are structured, but by defining how they communicate with each other.

The Crystal Reports Automation Server is available for both 16 and 32 bit (Cpeaut16.dll/Cpeaut32.dll) environments. The cpeautXX.dll is the file that provides the COM interface to the Report Engine. It is a registered file and is referenced by its ClassID – "Crystal.CRPE.Application"

Development using the Automation Server requires that you first learn the Object Hierarchy. Developers familiar with other common Object Hierarchies such as Data Access Object, Excel and Outlook will find programming with the Crystal Reports Automation Server very easy. However, a good understanding of the structure of a Crystal Report is needed to adequately program all the

different levels of objects available. On a learning curve, this places the Automation Server between the OCX and Print Engine API.

OCX developers are often interested in the Automation Server because it provides access to over 500 properties and methods without having to deal with the complexity of direct API calls. For Print Engine API developers, the Automation Server is attractive because it adds additional functionality not normally available to them. The additional functionality includes GET and SET capabilities for fonts, cursor tracking, window and report events.

## Report Design Component

The Report Designer Component (RDC), is the newest addition to the Crystal Reports family. The RDC provides an integrated and powerful solution for VB developers. It is an ActiveX designer object that packs the reporting power of Crystal Reports 6 into a lightweight add-in for VB 5.0. This allows developers to open, design and customize reports within the VB design environment. Intuitive Report Experts make connecting to data and integrating powerful reports into applications flexible and efficient.

With hundreds of report properties, methods and events, developers have complete control over their report designs, using familiar Visual Basic code. Report distribution is simplified through a small component count and free runtime. Reports can be packaged within the Applications executable or stored outside the application in the traditional (.rpt) format.

The RDC has an extensive object model, which is comparable to the Crystal Reports 6 Automation Server. However, since the RDC components are native COM with a dual interface, there is no need for wrappers around the Print Engine. For developers who are familiar with the Automation Server, the transition is an easy one. Developers who are new to Crystal Reports will discover that the learning curve for RDC is made easier with a well-designed and flatter Object Model.

A major difference between the RDC and the other developer tools within Crystal Reports is the ability to write formatting code in VB. The RDC provides event and callback support which allows developers to control the appearance of fields. The RDC also allows access to the values of individual records as they are processed. This new technology allows you to perform such activities as loading pictures from a file or conditionally formatting a field or a section at runtime.

The Crystal Smart Viewer replaces the Crystal Reports Preview Window as the means of previewing reports. This approach gives developers greater control and flexibility over the viewed report. The Smart Viewer is a lightweight ActiveX control that contains an extensive object and provides easy-to-use callback support. The viewer is customizable and allows you to add a custom tool bar if necessary.

In conclusion, Visual Basic developers who wish to run reports from an application have several methods to choose from. The Crystal ActiveX Control (OCX) is the simplest to use while the Print Engine API is the most complicated. In terms of ease of use, the Crystal Reports Automation Server and the Report Designer Component lies between the OCX and the Print Engine

API. The method you choose is dependent on your knowledge of Crystal Reports and Visual Basic, as well as your reporting requirements.

## Contacting Crystal Decisions for Technical Support

We recommend that you refer to the product documentation and that you visit our Technical Support web site for more resources.

**Self-serve Support:**

<http://support.crystaldecisions.com/>

**Email Support:**

<http://support.crystaldecisions.com/support/answers.asp>

**Telephone Support:**

<http://www.crystaldecisions.com/contact/support.asp>