VCL for .NET Overview

VCL for .NET is the programming framework for building Delphi 2005 applications using VCL components. Delphi 2005 and VCL for .NET are intended to help users leverage the power of Delphi when writing new applications, as well as for migrating existing Win32 applications to the .NET Framework.

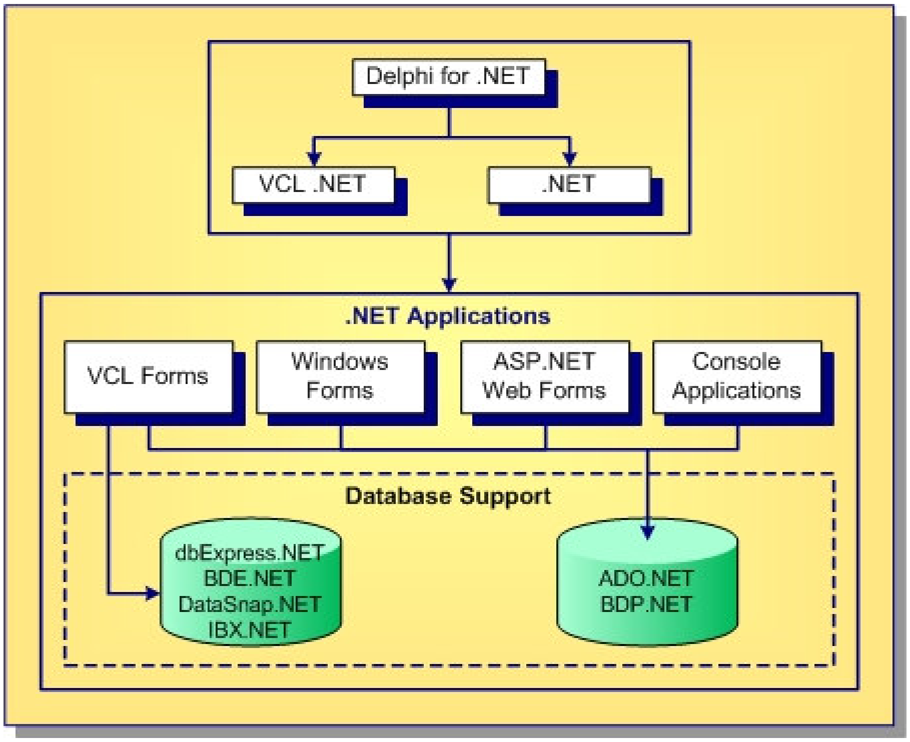
These technologies allow a Delphi developer to migrate to .NET, taking their Delphi skills and much of their current Delphi source code with them. Delphi 2005 supports Microsoft .NET Framework development with the Delphi language and both VCL for .NET controls and Windows Forms controls. Delphi 2005 ASP.NET also supports WebForms, and SOAP and XML Web Services application development.

VCL for .NET is a large subset of the most common classes in VCL for Win32. The .NET Framework was designed to accommodate any .NET-compliant language. In many cases Delphi source code that operates on Win32 VCL classes and functions recompiles with minimal changes on .NET. In some cases, the code recompiles with no changes at all. VCL for .NET is a large subset of VCL, therefore it supports many of the existing VCL classes. However, source code that calls directly to the Win32 API requires source code changes. Also, dependent third-party Win32 VCL controls need to be available in .NET versions for compatibility.

This section introduces:

- VCL for .NET Architecture
- VCL for .NET and the .NET Framework
- VCL for .NET Components
- Borland.VCL Namespace
- Porting Delphi Applications to Delphi 2005
- Importing .NET Components for Use in VCL for .NET Applications

VCL for .NET Architecture



VCL is a set of visual components for building Windows applications in the Delphi language. VCL for .NET is the same library of components updated for use in the .NET Framework. VCL for .NET and the .NET

Framework coexist within Delphi 2005. Both VCL for .NET and .NET provide components and functionality that allow you to build .NET applications:

- VCL for .NET provides the means to create VCL Forms applications, which are Delphi forms that are .NET-enabled, and use VCL for .NET components.
- VCL for .NET provides VCL non-visual components which have been .NET-enabled to access databases. You can also access databases through the ADO.NET and BDP.NET providers.
- .NET provides the means to build .NET Windows Forms, Web Forms, and Console applications, using .NET components, with Delphi code-behind.

You can build VCL Forms applications using VCL for .NET components, or Windows Forms applications using .NET components. You can also build ASP.NET Web Forms applications using either VCL for .NET components or .NET components.

VCL for .NET and the .NET Framework

The .NET Framework provides a library of components, classes, and low-level functionality that manages much of the common functionality, from the display of buttons to remoting functionality, without regard to the underlying implementation language. VCL for .NET and the .NET Framework are functionally equivalent. Like the .NET Framework, VCL for .NET provides libraries of components, controls, classes, and low-level functionality that help you build Windows Forms, Web Forms, and console applications that run on the current Windows .NET Framework platform.

VCL for .NET is not a replacement for the .NET Framework.

You will still need the .NET runtime to use VCL for .NET, but you can build complete applications using VCL for .NET components that will run on .NET platform.

You can build Delphi 2005 applications without using VCL for .NET, by creating Windows Forms, Web Forms, and Console applications using Delphi 2005 code.

You can use Delphi 2005 to create powerful .NET applications using .NET components, or VCL for .NET components that have been migrated from the Delphi VCL. If you have existing Delphi VCL applications that you want to run on Windows XP, you can easily port those applications by using Delphi 2005.

VCL for .NET Components

VCL for .NET consists of a set of visual and non-visual components. VCL for .NET builds on the concept of constructing applications visually, eliminating much manual coding.

Visual Components

Delphi 2005 provides a set of visual components, or controls, that you can use to build your applications. In addition to the common controls, such as buttons, text boxes, radio buttons, and check boxes, you can also find grid controls, scroll bars, spinners, calendar objects, a full-featured menu designer, and more. These controls are represented differently in Delphi 2005 than they are in frameworks, such as the .NET Framework.

In an IDE for other languages, such as C# or Java, you will see code-centric representations of forms and other visual components. These representations include physical definitions, such as size, height, and other properties, as well as constructors and destructors for the components. In the **Code Editor** of Delphi 2005 you will not see a code representation of your VCL for .NET components.

Delphi 2005 is a resource-centric system, which means that the primary code-behind representations are of event handlers that you fill in with your program logic. Visual components are declared and defined in text files with the extensions `.dfm` (Delphi Forms) or .`nfm` (Delphi 2005 forms). The `nfm` files are created by Delphi 2005 as you design your VCL Forms on the Forms Designer, and are listed in the resource list in the Project Manager for the given project.

Non-Visual Components

You can use non-visual components to implement functionality that is not necessarily exposed visually. For example, you can access data by using non-visual components like the BDP.NET components, which provide database connectivity and DataSet access. Even though these components do not have visual runtime behavior, they are represented by components in the **Tool Palette** at designtime. VCL for .NET provides a variety of non-visual components for data access, server functions, and more.

Borland.VCL Namespace

VCL for .NET classes are found under the `Borland.Vcl` namespace. Database-related classes are in the `Borland.Vcl.DB` namespace. Runtime library classes are in the `Borland.Vcl.Rtl` namespace.

Unit files have been bundled in corresponding `Borland.Vcl` namespaces. In some cases, units have been moved. However, namespaces are identified in a way that will assist you in finding the functionality you want.

Source files for all of the Delphi 2005 objects are available in the `c:\Program Files\Borland\BDS\3.0\Source` subdirectory.

Porting Delphi Applications to Delphi 2005

If you have existing applications written with an earlier version of Delphi, you might want to port them to .NET. In most cases, this will be easier than rewriting the applications. Because Delphi 2005 takes advantage of significant structural elements in the .NET Framework, you will need to perform some manual porting tasks to make your applications run. For example, the .NET Framework does not support pointers in safe code. So, any instance of a pChar or pointer variable will need to be changed to a .NET type. Many Delphi objects have been updated to accommodate these type restrictions, but your code may include references to pointers or unsupported types. For more information, refer to the Language Guide in this Help system.

Importing .NET Components for Use in VCL for .NET Applications

Delphi 2005 provides the **.NET Import Wizard** to help you import .NET controls into VCL for .NET units and packages. For example, you can wrap all .NET components, like those from the `System.Windows.Forms` assembly, in ActiveX wrappers that can be deployed on VCL for .NET applications. Once you have imported the .NET components of your choice, you can add a completed package file containing the units for each component to the **Tool Palette**. You can also view and modify the individual unit files, which can be useful reference material when you are writing your own custom components.

Related Information

Language Guide
Porting VCL Applications
Borland Data Providers for .NET
Importing .NET Controls to VCL for .NET
Deploying Applications
Building a VCL for .NET Form Application