What’s New in Raize Components 6

Raize Components is a *user interface design system* for Delphi and C++Builder. At its center is a collection of more than 125 general-purpose native VCL controls. Built on a foundation of technology first created more than fifteen years ago, these high-quality components give developers unsurpassed power and flexibility without sacrificing ease-of-use. In addition to the core set of controls, Raize Components includes more than 100 component designers focused on simplifying user interface development. Now more than ever, developers use Raize Components to build sophisticated user interfaces in less time with less effort.

The primary focus of Raize Components 6 is on the new VCL capabilities that have been added to **Embarcadero RAD Studio XE2**. Specifically, **64-bit VCL support** and the really cool new **VCL Styles** support. Unlike other releases of RAD Studio which simply required recompiling the existing code base to provide a new set of DCUs and packages, there were a lot of low-level changes made to the VCL in XE2. For example, many controls migrated smoothly to 64-bit, but many others required low-level modifications. In addition, every control in Raize Components was affected by the new VCL Styles. Each control's display code needed to be significantly modified to fully support for VCL Styles when they are available. It has been a lot of work, but we are certain that you will find the end result really cool!

There are other enhancements and fixes included in Raize Components 6, but 64-bit and VCL Styles support are the major new features. Raize Components 6 definitely leverages RAD Studio XE2, and to get the most out of RC6, you need XE2.

The following sections provide a more detailed look at what's new in Raize Components 6:

- Component Enhancements
- Package Changes
- Migration Issues
- Known Problems

Component Enhancements

TRzBorder

- Made necessary modifications to **TRzBorder** to fully support VCL Styles introduced in RAD Studio XE2.

TRzButton, TRzBitBtn, TRzMenuButton, TRzBmpButton

- Made necessary modifications to **TRzButton** and descendants to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzButton** and descendants to support 64-bit development.

TRzButtonEdit & TRzDBButtonEdit

- Cleaned up the appearance of embedded buttons in **TRzButtonEdit** under Windows Vista and Windows 7.
- Made necessary modifications to **TRzButtonEdit** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzButtonEdit** to support 64-bit.

TRzCalculator

- Made necessary modifications to **TRzCalculator** to fully support VCL Styles introduced in RAD Studio XE2.

TRzCalendar

- Fixed display positioning of separator lines on the **TRzCalendar**.
- Made necessary modifications to **TRzCalendar** to fully support VCL Styles introduced in RAD Studio XE2.

TRzCheckBox, TRzDBCheckBox & TRzRadioButton

- Fixed issue where incorrect custom glyphs was displayed for a **TRzCheckBox** using the grayed state and has the **HotTrack** property set to **False**;
- Fixed issue in **TRzCheckBox** and **TRzRadioButton** where the **WordWrap** property would get automatically set to True when **CustomGlyphs** were used.
- Made necessary modifications to **TRzCheckBox** and **TRzRadioButton** to fully support VCL Styles introduced in RAD Studio XE2. Including support for custom glyphs even when VCL Styles are used.

TRzCheckList

- Made necessary modifications to **TRzCheckList** to fully support VCL Styles introduced in RAD Studio XE2. Including support for custom check box glyphs.
- Made necessary modifications to **TRzCheckList** to support 64-bit.

TRzColorEdit

- Made necessary modifications to **TRzColorEdit** to fully support VCL Styles introduced in RAD Studio XE2.

TRzColorPicker

- Made necessary modifications to **TRzColorPicker** to fully support VCL Styles introduced in RAD Studio XE2.

TRzComboBox & TRzDBComboBox

- Fixed display issue in **TRzComboBox** and descendants where the under Vista and Windows 7, the background behind the dropdown arrow would not get changed if the **Color** property was changed and **FlatButtons** was **True**.
- Fixed display issue under Vista and Windows 7 in **TRzComboBox** where the standard themed control border would still be visible when using *Custom Framing* (i.e. **FrameVisible** := **True**) and removing one or more sides.
- Fixed issue in **TRzComboBox** where **Values** entries needed to be unique in order to properly locate an item. Now, multiple items in the list may be set to the same "value". Selecting an item with a duplicate value -- no longer -- results in the first item with that value from being automatically selected.
- Fixed issue where the dropdown list would not be sized appropriately if the combo box was set to use a larger font size.
- Made necessary modifications to **TRzComboBox** and **TRzDBComboBox** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzComboBox** and descendants to support 64-bit development.

TRzDateTimeEdit & TRzDBDateTimeEdit

- Added new **TwoDigitYearConverter** property to **TDBRzDateTimeEdit**, which controls how 2-digit years are converted to 4-digit years. The default value is **ycStandard**, which means that 2-digit years are converted using the **TwoDigitYearCenturyWindow** value. The **ycPastNotFuture** value means that a date with a 2-digit year is interpreted as a date in the past rather than the future. For example, consider entering '3/15/20' (in the U.S.). With the **ycStandard** converter value, the date gets converted to '3/15/2020'. But with the **ycPastNotFuture** converter, the converted date is '3/15/1920'. Entry of 4-digit years are that unaffected by this property.
- Added **PopupHeight** and **PopupWidth** properties to **TRzDBDateTimeEdit**. When both of these properties are set to non-zero values, the popup control (i.e. **Calendar** or **TimePicker**) is sized to the **PopupHeight** and **PopupWidth** values instead of the default auto size values.
- Made necessary modifications to **TRzDateTimeEdit** and **TRzDBDateTimeEdit** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzDateTimeEdit** and **TRzDBDateTimeEdit** to support 64-bit development.

TRzDBGrid

- Made necessary modifications to **TRzDBGrid** to fully support VCL Styles introduced in RAD Studio XE2—including automatic color adjustment for alternate rows when the **AltRowShading** property is set to True.
- Made necessary modifications to **TRzDBGrid** to support 64-bit development.

TRzDBLookupComboBox

- Fixed display issue with the dropdown button in **TRzDBLookupComboBox** when running under Windows Vista or later.
- Made necessary modifications to **TRzDBLookupComboBox** to fully support VCL Styles introduced in RAD Studio XE2.

TRzDBNavigator

- Made necessary modifications to **TRzDBNavigator** to fully support VCL Styles introduced in RAD Studio XE2. Including displaying the navigation images defined in the VCL Style.

TRzEdit & TRzDBEdit

- Fixed issue where the **TextHint** property would not display correctly in **TRzEdit** (and descendants) if the **FrameVisible** property was set to True.
- Made necessary modifications to **TRzEdit** and **TRzDBEdit** to fully support VCL Styles introduced in RAD Studio XE2.

TRzExpandEdit & TRzDBExpandEdit

- Made necessary modifications to **TRzExpandEdit** and **TRzDBExpandEdit** to fully support VCL Styles introduced in RAD Studio XE2.

TRzFontComboBox

- Fixed display issue with the Preview Panel in the **TRzFontComboBox** when it is initially displayed.
- Made necessary modifications to **TRzFontComboBox** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzFontComboBox** to support 64-bit development.

TRzFontListBox

- Fixed positioning issue with the *Font Preview* panel in the **TRzFontListBox**.
- Fixed issue in **TRzFontListBox** and descendants where the **GroupFont** properties would get inadvertently reset to the base font.
- Made necessary modifications to **TRzFontListBox** to fully support VCL Styles introduced in RAD Studio XE2. Including automatic color adjustment for groups.
- Made necessary modifications to **TRzFontListBox** to support 64-bit development.

TRzFormState

- Fixed issue in **TRzFormState** that prevented original **FormCreate** event handler from firing, if the **TRzFormState** was disabled at startup.

TRzGroupBar

- Made necessary modifications to **TRzGroupBar** to fully support VCL Styles introduced in RAD Studio XE2. When VCL Styles are used, the **TRzGroupBar** utilizes style appropriate colors to colorize the groups depending on the **GroupBar** style (i.e. *CategoryView*, *TaskList*, *Outlook*).
- Made necessary modifications to **TRzGroupBar** to support 64-bit development.

TRzGroupBox

- Made necessary modifications to **TRzGroupBox** to fully support VCL Styles introduced in RAD Studio XE2. Including automatic color adjustment of frame/border styles.

TRzImageComboBox

- Made necessary modifications to **TRzImageComboBox** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzImageComboBox** to support 64-bit development.

TRzLabel & TRzURLLabel

- Made necessary modifications to **TRzLabel** and **TRzURLLabel** to fully support VCL Styles introduced in RAD Studio XE2.

TRzLEDDisplay

- Made necessary modifications to **TRzLEDDisplay** to fully support VCL Styles introduced in RAD Studio XE2.

TRzLine

- Made necessary modifications to **TRzLine** to fully support VCL Styles introduced in RAD Studio XE2. Including automatic color conversion of line color and caption color.

TRzListBox, TRzTabbedListBox, TRzEditListBox, TRzRankListBox

- Fixed issue in **TRzListBox** and descendants where the **GroupFont** properties would get inadvertently reset to the base font.
- Fixed issue in **TRzListBox** and descendants where under very specific circumstances, the horizontal scroll bar may not be displayed even though **HorzScrollBar** is set to **True** and the width of the list box is not wide enough to display the full text of one or more items.
- Made necessary modifications to **TRzListBox**, **TRzTabbedListBox**, **TRzEditListBox**, and **TRzRankListBox** to fully support VCL Styles introduced in RAD Studio XE2. Including automatic color adjustment for groups.
- Made necessary modifications to **TRzListBox** (and descendants) to support 64-bit development.

TRzListView

- Redesigned the header display painting methods used by the **TRzListView**.
- Redesigned the code used to display the sort indicators in the headers.
- Made necessary modifications to **TRzListView** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzListView** to support 64-bit development.

TRzMemo & TRzDBMemo

- Made necessary modifications to **TRzMemo** and **TRzDBMemo** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzMemo** and **TRzDBMemo** to support 64-bit development.

TRzMeter

- Made necessary modifications to **TRzMeter** to fully support VCL Styles introduced in RAD Studio XE2.

TRzNumericEdit & TRzDBNumericEdit

- Made necessary modifications to **TRzNumericEdit** and **TRzDBNumericEdit** to fully support VCL Styles introduced in RAD Studio XE2.

TRzPageControl & TRzTabControl

- Made necessary modifications to **TRzPageControl** and **TRzTabControl** to fully support VCL Styles introduced in RAD Studio XE2. Since the **TRzPageControl** and **TRzTabControl** support multiple tab styles, the controls do not use the tabs directly from the selected VCL Style. Instead, the controls automatically select appropriate colors from the selected VCL Style to display the tabs.
- Moved **TRzTabMenuButton.DisplayMenu** method to public section.
- The menu items in the drop down menu for a **TRzPageControl** and **TRzTabControl** now have the same Hint that is assigned to the corresponding tabsheet or tab. The Hint string can be accessed with the **Application.Hint** property in an **Application.OnHint** event handler.

TRzPanel

- Made necessary modifications to **TRzPanel** to fully support VCL Styles introduced in RAD Studio XE2. Including automatic color adjustment of frame/border styles.

TRzPathBar

- Made necessary modifications to **TRzPathBar** to fully support VCL Styles introduced in RAD Studio XE2.

TRzProgressDisplay

- Fixed issue in **TRzProgressDisplay** where embedded CRs (#13) would cause the last line to be displayed outside of the bounds of the control instead of forcing a scroll.
- Made necessary modifications to **TRzProgressDisplay** to fully support VCL Styles introduced in RAD Studio XE2.

TRzRadioGroup, TRzDBRadioGroup & TRzCheckGroup

- Made necessary modifications to **TRzRadioGroup** and **TRzCheckGroup** to fully support VCL Styles introduced in RAD Studio XE2.

TRzRichEdit & TRzDBRichEdit

- Made necessary modifications to **TRzMemo** and **TRzDBMemo** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzMemo** and **TRzDBMemo** to support 64-bit development.

TRzSeparator

- Fixed display issue in **TRzSeparator** where no separator would be displayed if **ShowGradient** was set to **False**.
- Made necessary modifications to **TRzSeparator** to fully support VCL Styles introduced in RAD Studio XE2.

TRzShellList

- Made necessary modifications to **TRzShellList** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to shell controls to support 64-bit.

TRzShellTree

- The **TRzShellTree** initializes the new **ItemHeightMargin** property inherited from the **TRzTreeView** component to a value of 3, which results in the **TRzShellTree** matching the spacing used by Windows for its shell tree in Explorer.
- Fixed memory leak in **TRzShellTree** that could occur if the control's window handled needed to be recreated at runtime. For example, changing the **BorderStyle** property at runtime causes the control's window handle to be recreated. Very few properties caused the window to be recreated. However, the window does get recreated when changing VCL Styles at runtime.
- The **RefreshNodes** and **SortNode** methods of **TRzCustomShellTree** are now dynamic methods.
- Made necessary modifications to **TRzShellTree** and **TRzShellCombo** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to shell controls to support 64-bit.

TRzSizePanel

- Made necessary modifications to **TRzSizePanel** to fully support VCL Styles introduced in RAD Studio XE2. Including automatic color adjustment of **HotSpot** bar when VCL Styles are used.

TRzSpinEdit & TRzDBSpinEdit

- Cleaned up the appearance of embedded buttons in **TRzDBSpinEdit** under Windows Vista and Windows 7.
- Made necessary modifications to **TRzDBSpinEdit** and **TRzDBSpinner** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzDBSpinEdit** and **TRzDBSpinner** to support 64-bit development.

TRzSplitter

- Made necessary modifications to **TRzSplitter** to fully support VCL Styles introduced in RAD Studio XE2. Including automatic color adjustment of **HotSpot** bar when VCL Styles are used.
- Made necessary modifications to **TRzSplitter** to support 64-bit development.

TRzStatusBar

- Made necessary modifications to **TRzStatusBar** to fully support VCL Styles introduced in RAD Studio XE2. Including automatic color adjustment of frame/border styles.

TRzStatusPane & descendants

- Made necessary modifications to **TRzStatusPane**, **TRzGlyphStatus**, **TRzClockStatus**, **TRzKeyStatus**, **TRzResourceStatus**, **TRzMarqueeStatus**, **TRzProgressStatus**, **TRzFieldStatus**, and **TRzVersionInfoStatus** to fully support VCL Styles introduced in RAD Studio XE2.
- Fixed issue where **TRzResourceStatus** would always display 0 KB Free Memory when **DisplayStyle** was set to **dsText**.

TRzStringGrid

- Added new **UseDrawingStyle** property to **TRzStringGrid**. This property only applies when using RAD Studio 2010 or higher. When it is **True** (the default) the appearance of the grid is controlled by the built-in **DrawingStyle** property. When **UseDrawingStyle** is set to False, the appearance of the grid is controlled by the various properties such as **LineColor**.
- Made necessary modifications to **TRzStringGrid** to fully support VCL Styles introduced in RAD Studio XE2.
- Made necessary modifications to **TRzStringGrid** to support 64-bit.

TRzTimePicker

- **TRzTimePicker** now displays correct themed radio buttons for AM and PM.
- Made necessary modifications to **TRzTimePicker** to fully support VCL Styles introduced in RAD Studio XE2.

TRzToolbar

- Fixed issue in **TRzToolbar** when setting the **Align** property to something other than **alTop** or **alBottom** and **AutoSize** to **True** would cause the drop-down menu button to appear and overlap the last button in the display.
- Added new **DisabledImages** property to the **TRzToolbar** component. This property allows a user to associate a dedicated image list to the control that only contains disabled images. The benefit of this is that the same **ImageIndex** value can be used for both the Images and **DisabledImages** lists. The **TRzToolButtons** that have images will use the **DisabledImages** list if it is not nil *and* the button's **DisabledIndex** property is -1. The **ImageIndex** property will be used to index into the **DisabledImages** list.
- Made necessary modifications to **TRzToolbar** to fully support VCL Styles introduced in RAD Studio XE2. Including automatic color adjustment of frame/border styles.
- Made necessary modifications to **TRzToolbar** to support 64-bit development.

TRzToolButton

- Fixed display issue in **TRzToolButton** where the button would be drawn flat (i.e. without a border) even if the Flat property was False and the application was compiled without Themes.
- Fixed display issue in **TRzToolButton** where the button would not appear pressed when a drop-down menu was displayed by the button.
- Added new **DisabledImages** property to the **TRzToolButton** component. This property allows a user to associate a dedicated image list to the control that only contains disabled images. The benefit of this is that the same **ImageIndex** value can be used for both the **Images** and **DisabledImages** lists. The **TRzToolButtons** that have images will use the **DisabledImages** list if it is not nil *and* the button's **DisabledIndex** property is -1. The **ImageIndex** property will be used to index into the **DisabledImages** list.
- Made necessary modifications to **TRzToolButton** to fully support VCL Styles introduced in RAD Studio XE2.

TRzTrackBar

- When XP/Vista/Win7 themes are used, the **TRzTrackBar** now displays the thumb as defined in the selected style. This change fixes the issue where the thumb would appear like a Windows XP even on Windows Vista or 7.
- Made necessary modifications to **TRzTrackBar** to fully support VCL Styles introduced in RAD Studio XE2.

TRzTrayIcon

- Updated code in **TRzTrayIcon** used to restore application window.

TRzTreeView & TRzCheckTree

- Fixed issue where changing the background color of a **TRzCheckTree** at runtime would not change the background color of the check state images.
- Added new **ItemHeightMargin** property that can be used to add some extra spacing between items in a **TRzTreeView** (or descendant).
- Made necessary modifications to **TRzTreeView** and **TRzCheckTree** to fully support VCL Styles introduced in RAD Studio XE2.

RzCommon Unit

- Added **UsingSystemStyle** function.
- Added **ActiveStyleSystemColor** and **GetStyleStyleColor** functions.
- Added new **DrawFilledBox** procedure.
- Added Color parameter to **DrawFocusBorder**.
- Made necessary modifications to **TRzMenuController** to support VCL Styles introduced in RAD Studio XE2. Standard menus and popup menus are not affected when applying VCL Styles. However, with the **TRzMenuController**, menu items are displayed using appropriate colors and shading from the selected VCL Style so that the menus match the rest of the interface.
- Modified several drawing functions to support VCL Styles introduced in RAD Studio XE2.

Package Changes

The packages distributed with **Raize Components 6** are listed below. The naming convention used in the packages is such that in all future versions of Raize Components, the names of the package DCP files will no longer need to be changed. As always, there are two sets of packages associated with each version of the VCL, one runtime/design package pair for all non-data-aware components and another runtime/design package pair for all data-aware components.

When building an application that uses runtime packages, you may need to deploy the Raize Components runtime packages with your application. The design-time packages are not redistributable.

The following table describes the new package BPL files:

Package	Pkg Type	RAD Studio	VCL	Contents
RaizeComponentsVcl120.bpl	Runtime	2009	12.0	Non-Data-Aware components
RaizeComponentsVcl140.bpl	Runtime	2010	14.0	Non-Data-Aware components
RaizeComponentsVcl150.bpl	Runtime	XE	15.0	Non-Data-Aware components
RaizeComponentsVcl160.bpl	Runtime	XE2	16.0	Non-Data-Aware components
RaizeComponentsVclDb120.bpl	Runtime	2009	12.0	Data-Aware components
RaizeComponentsVclDb140.bpl	Runtime	2010	14.0	Data-Aware components
RaizeComponentsVclDb150.bpl	Runtime	XE	15.0	Data-Aware components
RaizeComponentsVclDb160.bpl	Runtime	XE2	16.0	Data-Aware components
RaizeComponentsVcl_Design120.bpl	Design	2009	12.0	Raize Components 6.x
RaizeComponentsVcl_Design140.bpl	Design	2010	14.0	Raize Components 6.x
RaizeComponentsVcl_Design150.bpl	Design	XE	15.0	Raize Components 6.x
RaizeComponentsVcl_Design160.bpl	Design	XE2	16.0	Raize Components 6.x
RaizeComponentsVclDb_Design120.bpl	Design	2009	12.0	Raize Components 6.x (Data-Aware)
RaizeComponentsVclDb_Design140.bpl	Design	2010	14.0	Raize Components 6.x (Data-Aware)
RaizeComponentsVclDb_Design150.bpl	Design	XE	15.0	Raize Components 6.x (Data-Aware)
RaizeComponentsVclDb_Design160.bpl	Design	XE2	16.0	Raize Components 6.x (Data-Aware)

If you are building your own packages that are dependent on the Raize Components units, you will need to add the appropriate DCP package information file to the **requires** clause of your package. With the new naming convention, the name of the DCP file is same regardless of the version of the VCL.

The following table lists the new package DCP files.

DCP File	Contents
RaizeComponentsVcl.dcp	Non-Data-Aware package information
RaizeComponentsVclDb.dcp	Data-Aware package information

Migration Issues

Raize Components 6 is backward compatible with earlier versions of Raize Components. That is, if you created an application using an earlier version of Raize Components, you will be able to load your forms under version 6.

Package Changes

If you are migrating from Raize Components 4 or 5, there should be no changes necessary in your packages. Simply rebuild them with Raize Components 6 installed.

If you are using an older version of Raize Components, you will need to modify the **requires** clauses of your packages to reflect the new package names. (Please see the Package Changes section for more details). As a result, there are some potential migration issues for Delphi and C++Builder users.

Delphi Users

If you have an existing project that used an earlier version of Raize Components, the option file (*.dof) for that project will contain references to the Raize Components 3.0 or 2.x runtime packages. The DefProj.dof file in your Delphi Bin directory is automatically updated during the install program, but your existing DOF files are not modified.

IMPORTANT: The runtime package list in a project's DOF file is only used if the project is built with runtime packages. The following steps are not necessary if you are not using runtime packages for your project.

If you are using runtime packages, then you should open the DOF file and look for the Packages entry in the [Directories] section. There you will find the two runtime packages for Raize Components listed. The actual file names will depend on the version of Raize Components you had previous to 6.

Old Version	Old Runtime Package	Change to
2.5	Rz25NDnn	RaizeComponentsVcl
2.5	Rz25DBnn	RaizeComponentsVclDb
2.51	Rz251Nnn	RaizeComponentsVcl
2.51	Rz251Dnn	RaizeComponentsVclDb
2.52	Rz252Nnn	RaizeComponentsVcl
2.52	Rz252Dnn	RaizeComponentsVclDb
3.x	Rz30Ctlsnn	RaizeComponentsVcl
3.x	Rz30DBCtlsnn	RaizeComponentsVclDb

*nn represents VCL version number (e.g. 70 for VCL 7.0)

Simply change the runtime package names to the 6.0 versions also listed in the above table, and then save the project option file.

C++Builder Users

If you have an existing project that used an earlier version of Raize Components, the project (*.bpr) file for that project will contain references to the Raize Components 3.0 or 2.x package BPI files. The Default.bpr file in your Builder Bin directory is automatically updated during the install program, but your existing BPR files are not modified.

As a result, you should open the BPR file and look for the line starting with PACKAGES. There you will find the two BPI files for Raize Components. The actual file names will depend on the version of Raize Components you had previous to version 6.

Old Version	Old BPI Files	Change to
2.52	Rz252Nnn.bpi	RaizeComponentsVcl.bpi
2.52	Rz252Dnn.bpi	RaizeComponentsVclDb.bpi
3.0	Rz30Ctlsnn.bpi	RaizeComponentsVcl.bpi
3.0	Rz30DBCtlsnn.bpi	RaizeComponentsVclDb.bpi

*nn represents VCL version number (e.g. 70 for VCL 7.0)

Simply change the BPI file names to the 6.0 versions also listed in the above table, and then save the project file.