

Visual PRO/5 6.0 Gives Apps an XP or Vista Contemporary Look and Feel

By Nick Decker



At first glance, an operating system's theme may appear to be just another pretty face. However, this look and feel is more than skin deep – it defines how users interact with the operating system and its applications. End users become accustomed to the target look and feel (L&F), which provides comfort and familiarity. It makes applications more intuitive as certain visual styles can effectively relay at a glance the use and capability of a particular control.

Until recently, Visual PRO/5® applications were not able to take on the target operating system's L&F, making it appear antiquated and less intuitive to use. Visual PRO/5 6.0 changes all of that, automatically presenting any BBx® program in the modern L&F of the target operating system. No matter if the application is running on a current operating system, such as Windows XP, or an operating system of tomorrow such as Windows Vista, the application runs in the best possible light and fits right in with the rest of the operating system.

Visual PRO/5 and Themes

BBj® has supported themes from both Java and SkinLF for years. But what about themes for Visual PRO/5? Until now, the only way that developers could influence the L&F of their Visual PRO/5-based application was to add a raised or sunken border to a control. The client edge and raised edge options gave the GUI designers the ability to mimic the L&F of the time – the Windows 95 3D beveled L&F. There are a couple of very important distinctions about how BASIS implemented this in earlier versions of Visual PRO/5.

Firstly, the developer was in charge of determining which edge options the controls should have. That put the onus on the BASIS developers to create an application that adhered to the L&F. In other words, the L&F did not come through automatically – the developer essentially 'hard-coded' the appearance for each control. While this did not seem so egregious at the time since there was only one L&F, most programmers today shudder at the thought of hard coding anything, believing that hard coding is bound to get you into trouble sooner or later. As it turns out, hard coding the L&F was no exception.

Secondly, the client and raised edges were available to every control. That is a significant point as the Windows 95 L&F dictated that some controls had a beveled edge while others did not. Since Visual PRO/5 would add or remove edges without discernment, it enabled developers to easily bypass the conventions of the L&F and create non-standard GUIs. This was non-standard in the sense that Visual PRO/5 applications could misuse existing elements of the L&F, sometimes resulting in an odd and awkward looking interface that appeared out of place. This was not a good feature as it allowed the developer to introduce anomalies to the L&F that deviated from the norm. **Figure 1** demonstrates the effect, showing various combinations of client and raised edges of the edit and button controls.

To Visual PRO/5's credit, it gave the correct defaults to the controls. In the example in **Figure 1** the edit control has the client edge property 'on' by default while the button has no edges defined. Looking at this screenshot, the

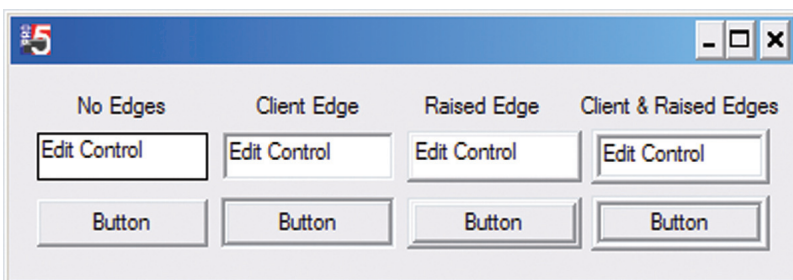


Figure 1. Visual PRO/5 and the various combinations of client and raised edges on controls

makes it look like its sitting on top of yet another button. If the Developer modified the default edges for controls, the resultant application ended up looking different from other Windows applications and could, in some cases, present a confusing interface to the user.

There is more!

Read this complete article online

for a closer look at skinning's beginnings, the impact of themes across other industries, and the progression to today's modern look and feel now available in Visual PRO/5 6.0. Go to

www.basis.com/advantage/mag-v10n1/supp.html

controls appear correct when rendered with these options; however they do not look correct when rendered with other options. Specifically, the edit control looks normal with the client edge, having the familiar Windows 95 sunken beveled look. When created with a raised edge, however, it 'pops out' of the form and looks more like a white button than an edit control. Likewise, the button looks correct without any edges. Adding a raised edge to it



Nick Decker
Engineering
Supervisor

continued...

Visual PRO/5 6.0 Adds Support for Themes

Visual PRO/5 programs, regardless of which version of Windows they ran on, always displayed in the ‘hard coded’ Windows 95 L&F. As far as the ‘look’ goes, a Visual PRO/5 application always used the Windows 95 beveled edges, even though Microsoft eliminated them in Windows XP and the upcoming Vista. In addition to presenting the wrong look for the operating system, the ‘feel’ part of the equation was missing as well. The hover effects, for example, were conspicuously absent. Today’s end users are now accustomed to the feedback provided by the operating system in subtle ways – fade and glow effects that result from hovering over a control, the change in a button’s appearance when it is the default button on the form or when it currently has focus, and so on. None of these effects were available because Visual PRO/5 did not automatically adapt to the current L&F.

Screenshots in the following figures demonstrate this concept using the button control. **Figure 2** shows what the button control looks like in the five possible states when theme support is disabled. In addition to having the older Windows 95 look, the ‘feel’ aspect is missing since the hovered state is the same as the normal state. In other words, when the user moves the mouse over the button, nothing happens.

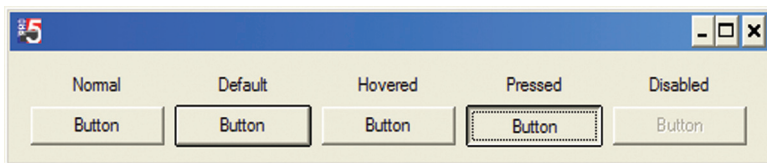


Figure 2. Visual PRO/5 buttons without theme support

Contrast this with **Figure 3** that shows the button controls in the same five states but themed in Windows XP. The look of the controls is very different, displaying round-cornered buttons with a slight gradient to add depth and colors to indicate state. The feel is very different as well – when

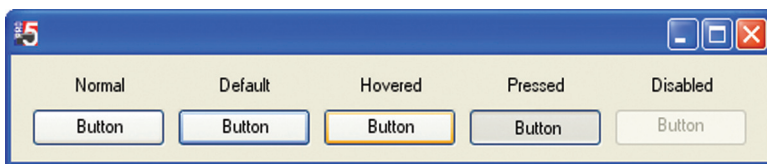


Figure 3. Visual PRO/5 buttons with theme support on Windows XP

moving the mouse over the button, the button changes its appearance to the hovered state of glowing orange. Taking the ‘feel’ aspect one step further in Vista (**Figure 4**),

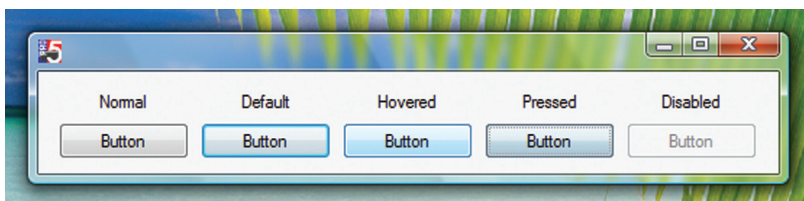


Figure 4. Visual PRO/5 buttons with theme support on Windows Vista Aero

the theme for a default button goes beyond a simple blue outline and actually pulsates, changing color in real time.

These effects are not just limited to buttons; they affect many of the other controls, such as the checkbox control. If Visual PRO/5 uses the old Windows 95 L&F shown in **Figure 5**, there would be only two possible states for a box – checked and unchecked. Furthermore, the appearance of the checkbox does not change when hovered or pressed. Running the same program under Windows XP or Windows Vista with theme support increases the number of states from two to six as illustrated in **Figures 6 and 7**; checked, unchecked, hovered checked and unchecked, and pressed checked and unchecked.

Automatic Theme Support

Visual PRO/5 6.0 solves the theme support problems of past revisions and introduces full-scale theme support. The first and probably most important feature of Visual PRO/5’s theme support is that it now implements automatically. BASIS developers do not have to change any code or modify any resources. Their applications will magically take on the L&F of its operating system’s theme. In other words, any legacy Visual PRO/5 application will run in Visual PRO/5 6.0 under three different operating systems – Windows 2000, Windows XP, and Windows Vista – and will display the user interface consistent with the L&F of the operating system. Not only will all of the controls display correctly with the appropriate border styles and so on, they will also behave correctly by exhibiting hover effects, rendering the default button state, etc.

The second feature of Visual PRO/5’s theme support is that of normalizing the appearance of controls with respect to their edges. As mentioned before, previous versions of Visual PRO/5 allowed developers to create non-standard controls by mixing and matching various edge styles. In order to comply fully with the target L&F, Visual PRO/5 6.0 would have to solve the problem of non-standard edges. Previous versions merely suggested that developers provide a consistent L&F by presenting suitable defaults. Visual PRO/5 6.0 had to be much more stringent than past versions to accomplish its ambitious goal so BASIS designed it to prohibit edges that failed to comply with the target L&F. As a result, it is possible to upgrade legacy applications instantly to a modern L&F without requiring any developer effort.

Backwards Compatibility

Normalizing the controls in this manner can have an unexpected side effect. For example, if a developer defines a control not to have either a client edge or a raised edge, but the target theme dictates that the control must have an edge, Visual PRO/5 will adjust the control accordingly to draw the appropriate edge. In most cases, the newly added edge appears as a few pixels on all sides of the control – seemingly hardly enough of a change to make a noticeable difference. However, a four-pixel adjustment in the control’s width and height must come from somewhere. The two places where this adjustment may occur are 1) outside of the control or the *non-client* area or 2) inside of the control or the *client* area. Visual PRO/5 makes the adjustment via the

continued...

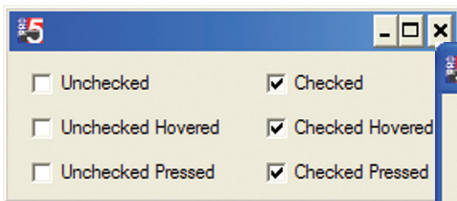


Figure 5. Unthemed checkbox controls

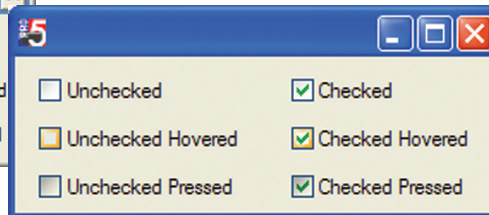


Figure 6. Themed checkbox controls on Windows XP

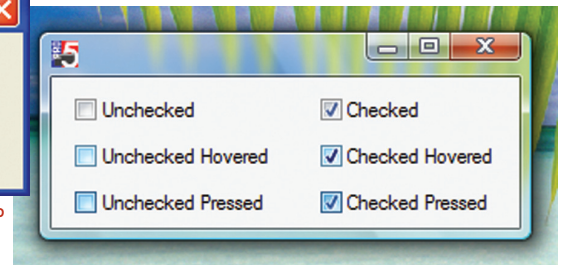


Figure 7. Themed checkbox controls on Windows Vista Aero

latter option – decreasing the size of the control’s client area in order to make room for the new edge. Option 2 is generally preferred, as the alternative would increase the size of the controls, possibly resulting in the controls overlapping one another on the form if they originally appeared in close proximity. Taking the space from the client area of the control prevents them from growing larger and reduces the number of possible harmful side effects.

However, option 2 is not without potential side effects since the content of the control now has a smaller container. The grid is a good example of this problem as it is an amalgamation of three separate controls – the grid itself, the row header, and the column header. Older versions of Visual PRO/5 strictly applied the edges to the grid control itself and completely ignored the headers. This led to a sort of disjointed control as the headers appeared to be floating separately from the grid. Most modern L&Fs apply the edge to the control as a whole – the grid and the two headers. Visual PRO/5 6.0 does this too when themes are turned on, but this leads to a discrepancy in how the grid is rendered compared to previous versions of Visual PRO/5. The result is that the grid’s client area may vary slightly with theme support turned on. In most instances, applications will not be susceptible to that minute difference. However, in case this does affect an application, BASIS created a keyword to appear in the configuration file to ensure backwards compatibility. Simply add the keyword **DISABLE_GRID_BORDER_THEME** to the **config.bbx** file to instruct Visual PRO/5 to enable all theme support except for grid borders. Other parts of the grid, such as the headers, will still be themed appropriately, but the edges will remain consistent with past versions to ensure correct sizing and placement of the grid control.

No doubt, most developers will love Visual PRO/5’s ability to change like a chameleon, automatically taking on the L&F of the operating system on which it is running. In the special case that requires the L&F of previous versions, a developer can bypass theme support all together by adding yet another new keyword, **DISABLE_THEME_SUPPORT**, to the **config.bbx** file. This forces Visual PRO/5 to revert to the legacy behavior, drawing all of the controls in the older style with the user-defined edges as shown in Figure 8.

Summary

Contemporary operating systems utilize themes to define the L&F of the operating system and its applications, customizing and enriching the user experience with specialized graphics and effects. BBx developers can now take advantage of these themes by running their applications in Visual PRO/5 6.0 which fully embraces the capabilities of the current L&F. No matter if the application is running on a current operating system such as Windows XP (Figure 9), an operating system of tomorrow such as Windows Vista (see Figure 10), or even

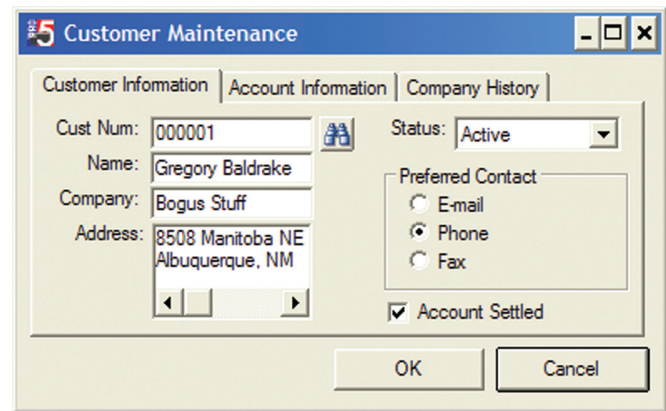


Figure 8. Sample application running non-themed

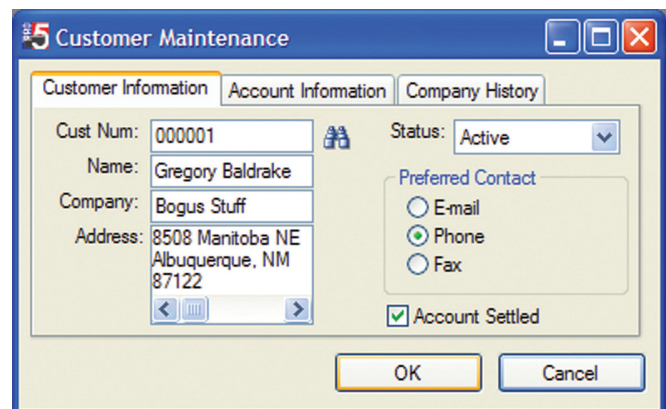


Figure 9. Sample application running with Windows XP Luna

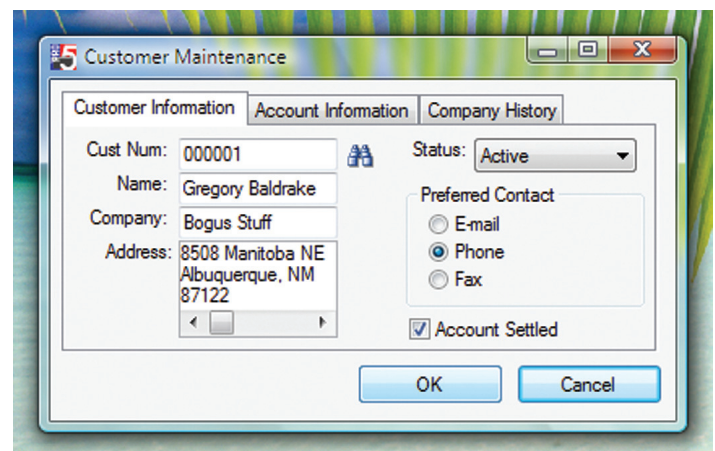


Figure 10. Sample application running with Windows Vista Aero

continued...

on an operating system utilizing third party themes (see **Figure 11**), the application runs in the best possible light and fits in with the rest of the operating system. 

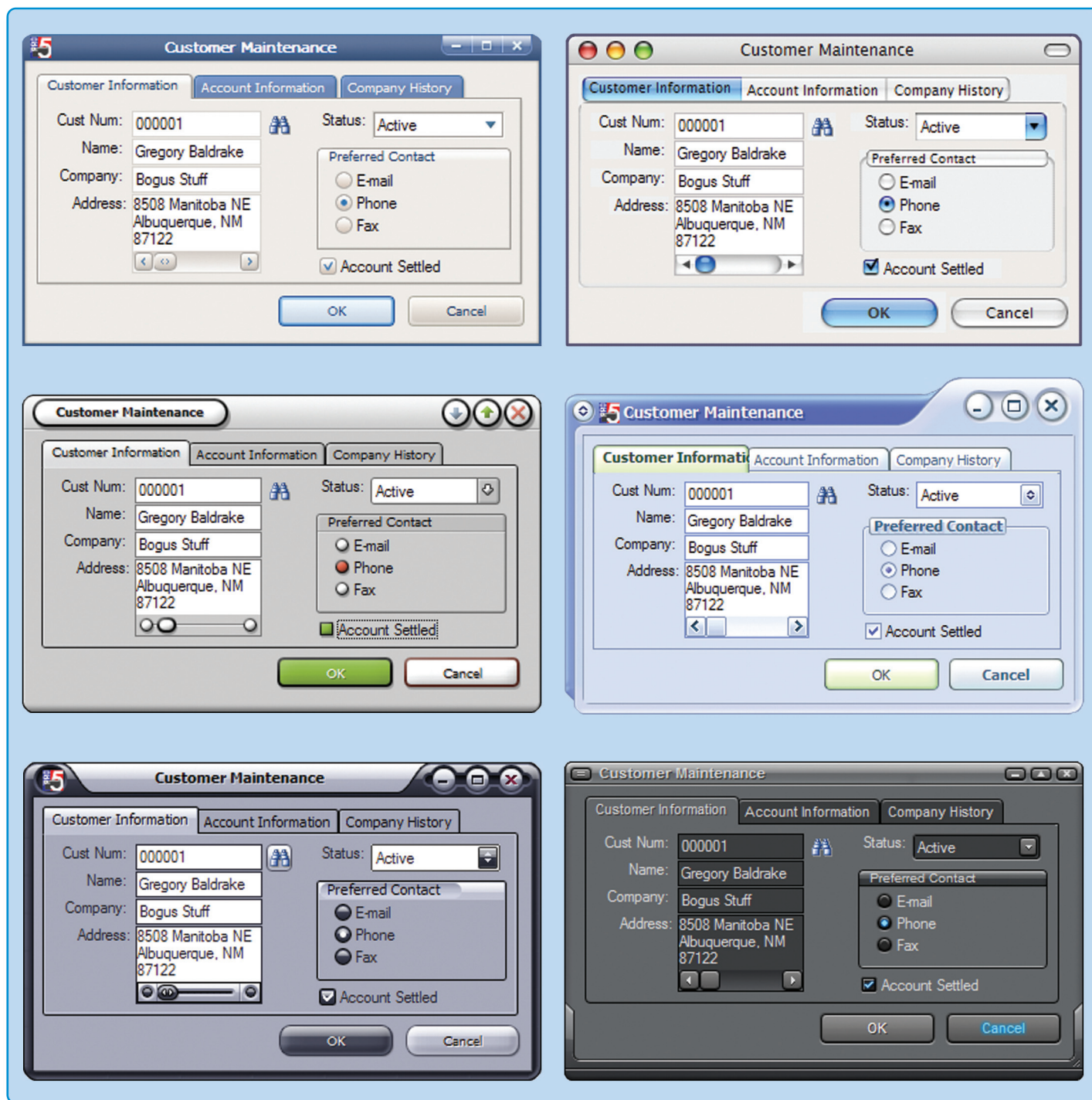


Figure 11. Sample application running a variety of third party themes



Download Visual PRO/5 6.0 today and try out the contemporary L&F. Go to www.basis.com/products/downloads.html

Read about the journey of themes from the first skins to customization across many industries in the e-Supplement at www.basis.com/advantage/mag-v10n1/supp.html

For additional information related to themes and skins, read *BBj 2.0: More Than One Way To Skin An App* at www.basis.com/advantage/mag-v6n1/skin.html

