[Home Page](#) [Novità](#) [Aiuto](#)

Applicare lo stile visuale di Windows XP

http://www.vbsimple.net/howto/ht_058.htm**Difficoltà:** ► 1 / 5

Con l'avanzamento degli anni, a cominciare dalla pubblicazione di Windows 95 lo stile grafico delle varie edizioni di Windows è andato piano piano raffinandosi, aggiungendo inizialmente lo smussamento (*smoothing*) dei caratteri e poi le ombreggiature. Tuttavia fino a Windows 2000 lo stile grafico era rimasto molto simile a quello del vecchio parente Windows 95.

L'arrivo di Windows XP ha introdotto la possibilità per un tema (una combinazione di colori e stili) di variare anche gli attributi delle finestre, come l'altezza, la forma e i colori delle singole finestre. Ormai da alcuni anni le moderne applicazioni utilizzano il look moderno di XP e sarebbe davvero un peccato non poter usufruire di questa caratteristica anche in progetti scritti in Visual Basic.

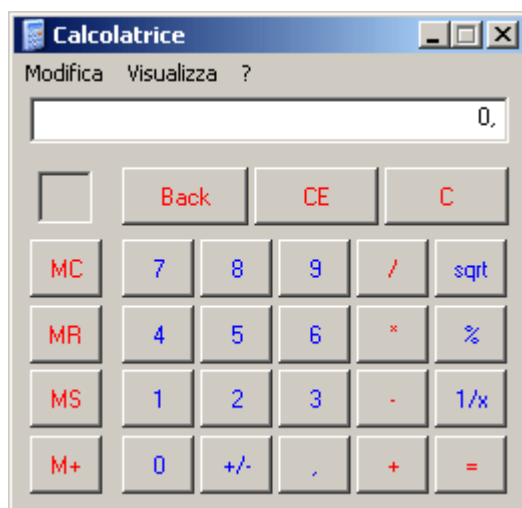


Figura 1

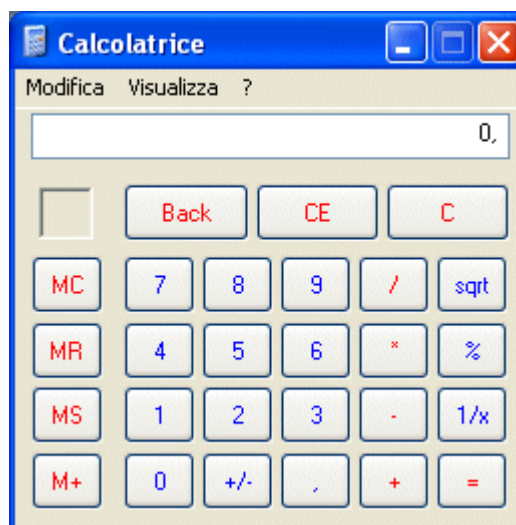


Figura 2

Le Figure 1 e 2 mostrano entrambe la Calcolatrice di Windows XP rispettivamente con il tema classico e con il tema di XP e le differenze sono parecchie: il colore di fondo è leggermente più chiaro, la barra e i pulsanti sul titolo sono più grandi e disegnati diversamente, la casella con il risultato nel tema XP non presenta il bordo incassato (bevel), i pulsanti hanno gli angoli arrotondati e tutta la finestra è circondata da un bordo blu. Ma le differenze non si limitano solo a questo e lo vedremo nel seguito di quest'articolo.

Prepareremo allo scopo un semplice progetto contenente quasi tutti i controlli **Thunder** (nativi di Visual Basic) e quasi tutti i controlli della libreria **Microsoft Windows Common Controls**. È necessaria una premessa riguardante proprio questi ultimi controlli.

Fino a Visual Basic 5 con il compilatore erano distribuiti anche due librerie [ActiveX](#)

([OCX](#)) di nome *COMCTL32.OCX* e *COMCT232.OCX* (chiamate informalmente **Common Controls 5**) contenenti all'interno tanti controlli standard di Windows presenti in quasi tutte le applicazioni moderne. Entrambe queste librerie si appoggiavano ad una terza libreria di sistema di nome *COMCTL32.DLL*. Il corretto funzionamento di un'applicazione che utilizzava uno di questi due OCX richiedeva la presenza della terza libreria aggiornata almeno alla versione 4.72. Questo spesso si traduceva nella necessità di dover distribuire un'altra libreria nel pacchetto di installazione. A questi tre files potevano aggiungersene altri per le singole localizzazioni (traduzioni del testo in esse contenuto).

Con l'arrivo di Visual Basic 6 gli sviluppatori Microsoft hanno avuto la brillante idea 😊 di riscrivere i suddetti OCX da capo includendo dentro le librerie *MSCOMCTL.OCX* e *MSCOMCT2.OCX* (chiamate informalmente **Common Controls 6**) il codice presente all'interno della libreria di sistema *COMCTL32.DLL*. Questo si traduceva nel vantaggio di non dover distribuire con il programma anche la terza libreria. Di contro naturalmente la dimensione delle due librerie è raddoppiata.

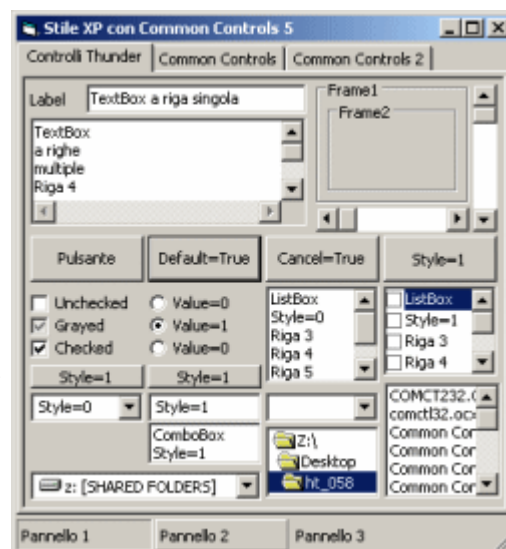
C'è da dire che con l'occasione gli sviluppatori hanno voluto migliorare alcuni controlli ed aggiungerne altri all'interno delle due librerie e questo, unito al vantaggio di non distribuire un file aggiuntivo nel pacchetto di installazione con il solo costo di un paio di centinaia di KB in più ha convinto parecchie persone. Tuttavia il problema era dietro l'angolo e l'avvento di Windows XP l'ha subito rivelato. 😞

Il lavoro svolto dagli sviluppatori di Windows XP infatti si basava sull'aggiornamento delle librerie di sistema fra cui anche *COMCTL32.DLL* e richiedeva la presenza di un file di configurazione che richiedesse al sistema l'applicazione del nuovo stile grafico. Questo ha comportato la possibilità di aggiornare con semplicità anche le applicazioni già compilate che utilizzassero i Common Controls 5, dovendo infatti aggiungere soltanto il file di configurazione esterno. Invece i programmi che utilizzano i Common Controls 6 hanno continuato ad applicare il vecchio stile grafico, non dovendo infatti appoggiarsi alla *COMCTL32.DLL*.

Acclusi a questo articolo saranno presenti due progetti: il primo utilizzerà i Common Controls 5 mentre il secondo utilizzerà i Common Controls 6; le differenze saranno minime e si limiteranno all'inclusione dei controlli presenti nella libreria più aggiornata. La barra del titolo mostrerà anche la versione dei controlli in uso.

Non saranno descritti i singoli componenti sul form perché poco importante e comunque troppo lungo; per semplificare la comprensione sono stati lasciati i nomi originali dei controlli fuorchè per la [matrice](#) di PictureBox **picGruppo** che conterrà al suo interno tutti gli altri controlli; è quindi proposto subito il codice del form uguali per i due progetti:

1. Option Explicit



```

2.
3. Private Declare Function InitCommonControls Lib "Comctl32.dll" () As Long
4.
5. Private Sub Form_Initialize()
6.     Call InitCommonControls
7. End Sub
8.

```







La funzione **InitCommonControls**, utilizzata durante l'avvio del programma (Initialize infatti è il primo evento che scatta durante l'avvio) provoca il caricamento in memoria della libreria *COMCTL32.DLL*. Sarà tale libreria ad occuparsi autonomamente dell'aggiornamento di tutti i controlli standard di VB e dei Common Controls 5. Quest'operazione è necessaria anche quando il progetto utilizza le librerie *COMCTL32.OCX* o *COMCTL32.DLL* e quindi indirettamente anche *COMCTL32.DLL*.

Basteranno infatti solo queste poche righe di codice per applicare lo stile visuale di Windows XP; tutto il resto del codice, molto semplice invero, si occuperà di utilizzare i numerosi controlli sul form.

```

9. Private Sub Slider1_Scroll(Index As Integer)
10.     ProgressBar1(Index).Value = Slider1(Index).Value
11. End Sub
12.
13. Private Sub TabStrip1_Click()
14.     If TabStrip1.SelectedItem.Index <> 1 Then picGruppo(0).Visible = False
15.     If TabStrip1.SelectedItem.Index <> 2 Then picGruppo(1).Visible = False
16.     If TabStrip1.SelectedItem.Index <> 3 Then picGruppo(2).Visible = False
17.     picGruppo(TabStrip1.SelectedItem.Index - 1).Visible = True
18. End Sub
19.
20. Private Sub Form_Load()
21.     TabStrip1_Click
22.     ProgressBar1(0).Value = ProgressBar1(0).Max \ 2
23.     ProgressBar1(1).Value = ProgressBar1(1).Max \ 2
24.     TreeView1.Nodes.Add TreeView1.Nodes.Add(, , , "ciao"), tvwChild, , "miao"
25.     ListView1(0).ListItems.Add , , "ciao"
26.     ListView1(0).ListItems.Add , , "miao"
27.     ListView1(1).ListItems.Add , , "ciao"
28.     ListView1(1).ListItems.Add , , "miao"
29. End Sub

```

Il codice funzionale non farà altro che legare le due ProgressBar  al valore dei due Slider  (righe 9-11); in funzione della scheda scelta nella TabStrip  sarà mostrata la PictureBox  corrispondente e verranno nascoste le altre due all'interno della stessa matrice **picGruppo** (righe 13-18); infine al caricamento del form saranno inseriti alcuni valori nella TreeView  e nelle due ListView  per vedere in funzione i suddetti controlli.

Se provassimo questo progetto su Windows XP non otterremmo completamente il nuovo stile visuale; verrà infatti aggiornata soltanto la parte esterna alla *client area*, la superficie del form e cioè vedremo aggiornata la barra del titolo ed i suoi pulsanti ed il bordino azzurro che circonda la finestra. Per funzionare correttamente manca ancora il file di configurazione dell'applicazione, il cosiddetto file **MANIFEST**. Si tratta di un semplice documento [XML](#) con uno schema ben preciso che dovrà esser posto nella stessa cartella del programma e con il nome del file eseguibile seguito dall'estensione .manifest.

Nel nostro esempio, avremo infatti ***Common Controls 5.exe*** ed il file di configurazione ***Common Controls 5.exe.manifest*** contenente la richiesta di utilizzare la versione 6 della libreria COMCTL32.DLL se disponibile:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">
  <assemblyIdentity
    name="Azienda.Prodotto"
    type="win32"
    version="1.0.0.0"
    processorArchitecture="X86"
  />
  <description>Descrizione dell'applicazione</description>
  <dependency>
    <dependentAssembly>
      <assemblyIdentity
        name="Microsoft.Windows.Common-Controls"
        type="win32"
        version="6.0.0.0"
        processorArchitecture="X86"
        publicKeyToken="6595b64144ccf1df"
        language="*"
      />
    </dependentAssembly>
  </dependency>
</assembly>
```

Il testo in grassetto può essere modificato con i dati della nostra applicazione: Nome, Versione e Descrizione. Al giorno d'oggi non esiste alcun controllo di congruità tra i dati dichiarati nel file manifest e quelli inseriti all'interno del programma eseguibile, ma in vista delle possibili modifiche alla tecnologia *side-by-side* su cui si basa questo sistema si raccomanda di fornire dati congruenti con quelli dell'applicazione.



Importante!

A differenza delle versioni 4 e 5 della libreria COMCTL32.DLL, la nuova versione 6 non è redistribuibile assieme all'applicazione, pertanto è necessario escluderla dal pacchetto di installazione del programma, che dovrà affidarsi alla versione già installata nella macchina dell'utente finale. Windows XP contiene solitamente al suo interno sia la versione 5 che la versione 6 della suddetta libreria, quindi non dovrebbe essere un problema.

Lanciamo l'esecuzione dei due progetti, il primo utilizzerà i Common Controls 5 mentre il secondo i Common Controls 6, entrambi su Windows XP e vediamo le differenze:



Figura 4



Figura 5

Le Figure 4 e 5 mostrano la prima scheda dei due programmi ed in entrambe possiamo vedere lo stile visuale di XP in azione mettendo a confronto i lati positivi con quelli negativi:

Pro 🏆

- Entrambe le caselle di testo presentano il bordino leggero e colorato;
- I tre pulsanti (**Pulsante**, **Default=True** e **Cancel=True**) presentano i bordi arrotondati;
- I due frames hanno il bordo arrotondato;
- Le due barre di scorrimento in alto a destra hanno un look più accattivante e moderno;
- Le prime 3 CheckBox ed i primi 3 OptionButton presentano il nuovo stile visuale;
- Le 3 ListBox ed i controlli DriveListBox, DirListBox e FileListBox hanno un'estetica aggiornata, con il bordo più snello e colorato e le barre di scorrimento differenti;

Contro 🐹

- Il quarto pulsante (**Style=1**) continua a mantenere il vecchio stile;
- Il testo del secondo frame, contenuto all'interno del primo ha perso la formattazione originale e risulta visibile solo in parte;
- La quarta CheckBox e il quarto OptionButton, entrambi con proprietà **Style=1** (Graphical) mantengono il vecchio stile grafico;



Figura 6

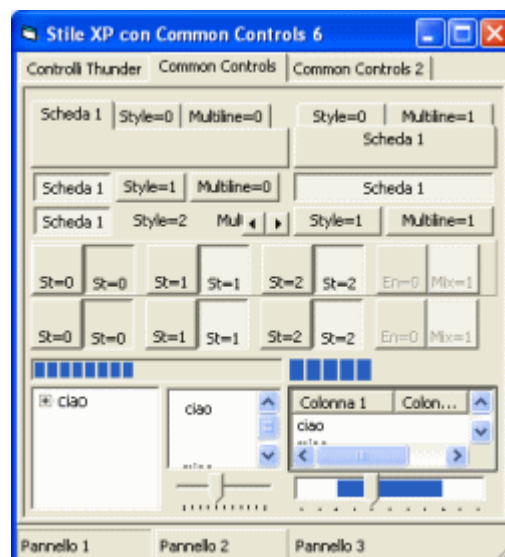


Figura 7

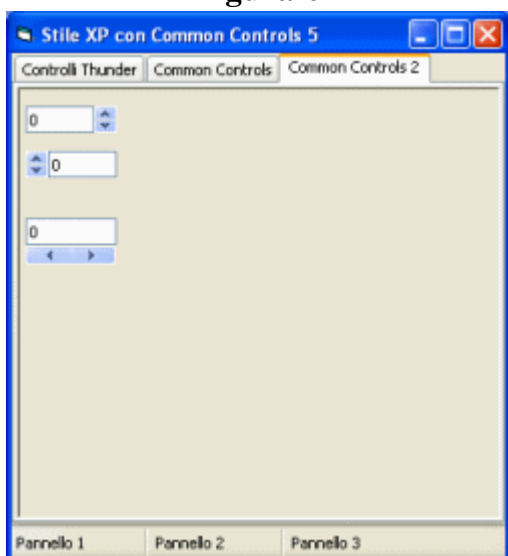


Figura 8

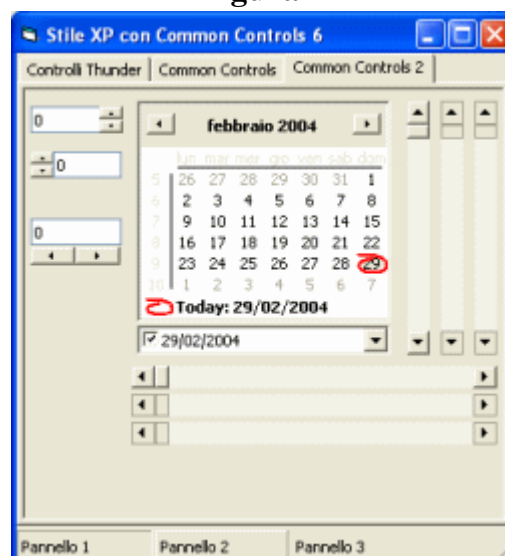


Figura 9

Nulla da dire invece per l'aspetto dei Common Controls; la versione 5 è perfettamente aggiornata (perché basata su *COMCTL32.DLL*) mentre la versione 6 a destra continua a mantenere il vecchio stile grafico, avendo infatti i programmatori inserito il codice dentro le librerie ActiveX anziché affidarsi alla *COMCTL32.DLL*.

Per quanto riguarda i controlli Thunder il nuovo stile è regolarmente applicato salvo che per quei controlli che presentano proprietà *Style=1 Graphical*; un bug grafico minore si verifica nel caso di frames annidati. Al di là dei controlli aggiuntivi mostrati nella **Figura 9**, i Common Controls 6 perdono nettamente questo confronto ed andrebbero evitati perché non in grado di aggiornare la loro estetica al nuovo stile di Windows XP.

Fibia.FBI

28 Febbraio 2004

Un ringraziamento a [GbCl](#) per il suo barbatrucco



[Torna all'indice degli HowTo](#)