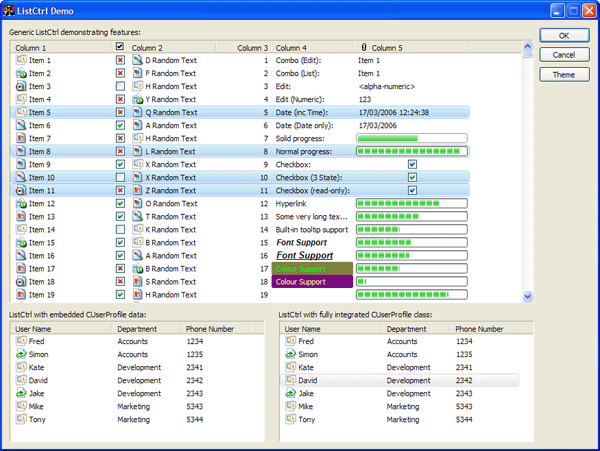
**ListCtrl - A WTL list control with Windows Vista style item selection**

By **[AlanW](http://65.39.148.34/script/Membership/View.aspx?mid=19066)** | 18 Apr 2006

[VC6](http://65.39.148.34/search.aspx?aidlst=4)[VC7](http://65.39.148.34/search.aspx?aidlst=5)[VC7.1](http://65.39.148.34/search.aspx?aidlst=7)[VC8.0](http://65.39.148.34/search.aspx?aidlst=9)[Windows](http://65.39.148.34/search.aspx?aidlst=94)[Visual-Studio](http://65.39.148.34/search.aspx?aidlst=103)[MFC](http://65.39.148.34/search.aspx?aidlst=107)[WTL](http://65.39.148.34/search.aspx?aidlst=109)[Dev](http://65.39.148.34/search.aspx?aidlst=118)[Intermediate](http://65.39.148.34/search.aspx?aidlst=153)

A flexible WTL list control that supports Windows Vista style selection and cell editing.



**Introduction**

Coming from an MFC background, I thought it was about time I embarked upon a new technological journey and learn how to develop fancy .NET applications, and impress everyone with my new found knowledge in C#. I've since made a great deal of progress on the trendy new platform, from learning about the internals of the CLR to the C# language syntax. However, I have one small niggling problem that won't go away. It doesn't excite me. Why it doesn't excite me is puzzling, since everyone I speak to and pretty much everything I read seems to regard the .NET framework as the best thing since someone had the bright idea of saving everyone the bother of slicing their own bread - maybe it's a control thing, or maybe I need to "let go".

Anyway, this rambling really belongs in another article; however, it serves as a good introduction into how I came into discovering WTL, and why I can't seem to kick the C++ habit in favour of fame and fortune in the world that is .NET. By the way, I should also point out that I love C++, and WTL does excite me.

**Background**

Whilst learning to use WTL, I decided I would begin by porting a list control I wrote a while ago using MFC and improve it along the way. This article is the fruit of my efforts - yet another list control that supports many features, but no doubt doesn't quite do everything you're looking for. But where's the fun in that!

**Features**

* Vista style item selection.
* Themed or classic UI support.
* Item or subitem selection.
* Subitem editing, combobox, and date/time.
* Hyperlinks, check boxes (normal and 3-state), and progress bars (themed and solid).
* Header column drag-drop re-ordering.
* Built-in support for column sorting.
* Groovy looking alpha-blending group select box.
* Double-buffered flicker free drawing (comes free with WTL).
* Smooth scrolling (with deceleration).
* Titletips (not to be confused with tooltips, but that's supported also).
* Everything looks jolly nice (not really a feature, but I thought I'd mention it).
* Embedded data structure or class support (I'll talk about this later).
* Data source overriding (I'll talk about this later too).

**Additional**

You may be interested to know that I also have a tree control that compliments the drawing style used inCListCtrl: [TreeCtrl - A WTL tree control with Windows Vista style item selection](http://www.codeproject.com/wtl/WTL_TreeCtrl.asp).

**How to use CListCtrl**

First of all, I think it would be a good idea to explain subitem and column formats / flags that can be used to control how text is displayed and edited in the list control. An item format may be set at column-level, meaning all subitems under that column share the same formatting, or at an individual subitem-level that will override the column format. An example of column/subitem formatting can be found in the demo where "Column 5" has a column-level format ofITEM\_FORMAT\_PROGRESS, however it's been overridden for individual subitems to demonstrate other formats.

I'd encourage you to play around with the demo; however, this being CodeProject, I'm guessing you've probably already done this. For those who would like a little more detail, here are the basic functions required to get you going using the generic CListCtrl:

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

void SetImageList(

CImageList& ilItemImages

)

**Description:**

Assigns an image list for list control items.

**Parameters:**

* ilItemImages - Reference to image list.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

void AddColumn(

CListColumn& listColumn

)

void AddColumn(

LPCTSTR lpszText,

int nWidth = 0,

int nImage = ITEM\_IMAGE\_NONE,

BOOL bFixed = FALSE,

UINT nFormat = ITEM\_FORMAT\_NONE,

UINT nFlags = ITEM\_FLAGS\_NONE

)

**Description:**

Adds a column to the list control.

**Parameters:**

* listColumn - Reference to a structure describing column details.
* lpszText - Column header title string.
* nWidth - Width of column in pixels.
* nImage - Index to imagelist to display in the column header.
* bFixed - TRUE for fixed column width.
* nFormat - Describes the format of all subitems in the column:
  + ITEM\_FORMAT\_NONE - Default, read-only subitem.
  + ITEM\_FORMAT\_EDIT - Editable subitem.
  + ITEM\_FORMAT\_DATETIME - Date/time subitem.
  + ITEM\_FORMAT\_COMBO - Combobox subitem.
  + ITEM\_FORMAT\_CHECKBOX - 2-state checkbox subitem.
  + ITEM\_FORMAT\_CHECKBOX\_3STATE - 3-state checkbox subitem.
  + ITEM\_FORMAT\_HYPERLINK - Hyperlink (clickable) subitem.
  + ITEM\_FORMAT\_PROGRESS - Progress indicator subitem.
  + ITEM\_FORMAT\_CUSTOM - Custom subitem.
* nFlags - Bit mask that describes the type of column format:
  + ITEM\_FLAGS\_NONE - No format flags (default).
  + ITEM\_FLAGS\_LEFT - Left justify column header and subitems text.
  + ITEM\_FLAGS\_RIGHT - Right justify column header and subitems text.
  + ITEM\_FLAGS\_CENTRE - Centre column header and subitems text.
  + ITEM\_FLAGS\_READ\_ONLY - Subitems in column are read only (useful forITEM\_FORMAT\_CHECKBOX[\_3STATE] or ITEM\_FORMAT\_DATETIME).
  + ITEM\_FLAGS\_EDIT\_UPPER - Edit text is forced to uppercase (ITEM\_FORMAT\_EDIT orITEM\_FORMAT\_COMBO + ITEM\_FLAGS\_COMBO\_EDIT).
  + ITEM\_FLAGS\_EDIT\_NUMBER - Edit text is numerical (ITEM\_FORMAT\_EDIT or ITEM\_FORMAT\_COMBO +ITEM\_FLAGS\_COMBO\_EDIT).
  + ITEM\_FLAGS\_EDIT\_FLOAT - Edit text is floating-point (ITEM\_FORMAT\_EDIT or ITEM\_FORMAT\_COMBO+ ITEM\_FLAGS\_COMBO\_EDIT).
  + ITEM\_FLAGS\_EDIT\_NEGATIVE - Edit text allows negative numbers (ITEM\_FLAGS\_EDIT\_NUMBER orITEM\_FLAGS\_EDIT\_FLOAT).
  + ITEM\_FLAGS\_EDIT\_OPERATOR - Numeric text can be preceded by ">", "<=", etc. (ITEM\_FLAGS\_EDIT\_NUMBER or ITEM\_FLAGS\_EDIT\_FLOAT).
  + ITEM\_FLAGS\_COMBO\_EDIT - Combobox contains edit control (ITEM\_FORMAT\_COMBO).
  + ITEM\_FLAGS\_DATE\_ONLY - Only date is shown or editable (ITEM\_FORMAT\_DATETIME).
  + ITEM\_FLAGS\_TIME\_ONLY - Only time is shown or editable (ITEM\_FORMAT\_DATETIME).
  + ITEM\_FLAGS\_DATETIME\_NONE - Allow "None" in date/time date picker (ITEM\_FORMAT\_DATETIME).
  + ITEM\_FLAGS\_PROGRESS\_SOLID - Progress bar is drawn in solid style (ITEM\_FORMAT\_PROGRESS).

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

int AddItem(

CListItem< TData >& listItem

)

int AddItem(

LPCTSTR lpszText,

int nImage = ITEM\_IMAGE\_NONE,

UINT nFormat = ITEM\_FORMAT\_NONE,

UINT nFlags = ITEM\_FLAGS\_NONE

)

**Description:**

Adds an item to the list control.

**Parameters:**

* listItem - Reference to a structure describing all item details.
* lpszText - First subitem text.
* nImage - Index to imagelist to display in first subitem (see SetImageList).
* nFormat - Describes format of first subitem (see AddColumn for description).
* nFlags - Bit mask that describes the type of subitem format (see AddColumn for description).

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

BOOL SetItemText(

int nItem,

int nSubItem,

LPCTSTR lpszText

)

**Description:**

Set subitem text.

**Parameters:**

* nItem - Index to item (zero index).
* nSubItem - Index to subitem (zero index).
* lpszText - Subitem text.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

BOOL SetItemDate(

int nItem,

int nSubItem,

SYSTEMTIME& stItemDate

)

**Description:**

Set subitem date (for ITEM\_FORMAT\_DATETIME subitems).

**Parameters:**

* nItem - Index to item (zero index).
* nSubItem - Index to subitem (zero index).
* stItemDate - Reference to SYSTEMTIME structure.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

BOOL SetItemComboIndex(

int nItem,

int nSubItem,

int nIndex

)

**Description:**

Set subitem combobox item selection (for ITEM\_FORMAT\_COMBO subitems).

**Parameters:**

* nItem - Index to item (zero index).
* nSubItem - Index to subitem (zero index).
* nIndex - Index to combobox list to set.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

BOOL SetItemCheck(

int nItem,

int nSubItem,

int nCheckValue

)

**Description:**

Set subitem checkbox state (for ITEM\_FORMAT\_CHECKBOX and ITEM\_FORMAT\_CHECKBOX\_3STATE subitems).

**Parameters:**

* nItem - Index to item (zero index).
* nSubItem - Index to subitem (zero index).
* nCheckValue - Number representing checkbox state:
  + 0 - Unchecked (or undefined for 3-state).
  + 1 - Checked (or "tick" for 3-state).
  + -1 - "cross" for 3-state (not applicable for 2-state checkboxes).

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

BOOL SetItemImage(

int nItem,

int nSubItem,

int nImage

)

**Description:**

Set subitem image.

**Parameters:**

* nItem - Index to item (zero index).
* nSubItem - Index to subitem (zero index).
* nImage - Index to imagelist to display in subitem (see SetImageList).

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

BOOL SetItemFormat(

int nItem,

int nSubItem,

UINT nFormat,

UINT nFlags = ITEM\_FLAGS\_NONE

)

BOOL SetItemFormat(

int nItem,

int nSubItem,

UINT nFormat,

UINT nFlags,

CListArray < CString >& aComboList

)

**Description:**

Set subitem format (overrides column format).

**Parameters:**

* nItem - Index to item (zero index).
* nSubItem - Index to subitem (zero index).
* nFormat - Describes the format of subitem (see AddColumn for description).
* nFlags - Bit mask that describes the type of subitem format (see AddColumn for description).
* CListArray < CString >& aComboList - Array of strings that are used to populate combobox (forITEM\_FORMAT\_COMBO).

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

BOOL SetItemFont(

int nItem,

int nSubItem,

HFONT hFont

)

**Description:**

Set subitem font (overrides standard font).

**Parameters:**

* nItem - Index to item (zero index).
* nSubItem - Index to subitem (zero index).
* hFont - Handle to GDI font.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

BOOL SetItemColours(

int nItem,

int nSubItem,

COLORREF rgbBackground,

COLORREF rgbText

)

**Description:**

Set subitem background and text colours (overrides standard colours).

**Parameters:**

* nItem - Index to item (zero index).
* nSubItem - Index to subitem (zero index).
* rgbBackground - RGB value for subitem background.
* rgbText - RGB value for subitem text.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

void ShowThemed(

BOOL bShowThemed = TRUE

)

**Description:**

Override default header and selection drawing (applies to XP+).

**Parameters:**

* bShowThemed - TRUE = draw themed header and selection; FALSE = classic mode.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

void SetSmoothScroll(

BOOL bSmoothScroll = TRUE

)

**Description:**

Smooth scroll list when paging or line scrolling.

**Parameters:**

* bSmoothScroll - TRUE = smooth scroll; FALSE = normal scroll.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

void SetBackgroundImage(

HBITMAP hBackgroundImage,

BOOL bTileImage = FALSE

)

**Description:**

Set background bitmap image.

**Parameters:**

* hBackgroundImage - Handle to bitmap to set as background.
* bTileImage - TRUE = tile image; FALSE - draw image centered.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

void ShowHeader(

BOOL bShowHeader = TRUE

)

**Description:**

Show or hide list header.

**Parameters:**

* bShowHeader - TRUE = show list header; FALSE = hide list header.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

void ShowHeaderSort(

BOOL bShowSort = TRUE

)

**Description:**

Show sort (header) arrow and support column sorting.

**Parameters:**

* bShowSort - TRUE = support column sorting; FALSE = hide sort arrow.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

void SetSingleSelect(

BOOL bSingleSelect = TRUE

)

**Description:**

Only allow one selection (disable multiple and group selection).

**Parameters:**

* bSingleSelect - TRUE = single item selection only; FALSE = support multiple and group selection.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

void SetFocusSubItem(

BOOL bFocusSubItem = TRUE

)

**Description:**

Support subitem focus or entire item selection.

**Parameters:**

* bFocusSubItem - TRUE = support subitem focus; FALSE = entire item focus.

**How to use CListCtrlData< class TData >**

CListCtrl supports SetItemData and GetItemData; these set and retrieve DWORD values from list items that can be used for anything you like. However, you might find it more useful to integrate (or embed) your data structures or classes directly into the control. For this, you'll need the template class CListCtrlData (an example of this can be found in the demo application). The following functions can be used to access this data:

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

BOOL SetItemData(

int nItem,

TData& tData

)

**Description:**

Assigns data to an item.

**Parameters:**

* nItem - Index to item (zero index).
* tData - Reference to data structure or class (type defined in the template class declaration).

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

BOOL GetItemData(

int nItem,

TData& tData

)

**Description:**

Retrieves data from an item.

**Parameters:**

* nItem - Index to item (zero index).
* tData - Reference to data structure or class (type defined in template class declaration).

**How to use CListImpl**

Sometimes, it's not quite enough using the generic control. You may want to:

1. add new features to the control,
2. customise drawing, or
3. take total control over the list data source.

Since there's a clear separation of the interface and the data source built into the design of CListImpl, you can simply derive a class and override the data accessor functions, thereby taking total ownership of the data source including how the data is stored and accessed. In the demo, you'll find CListUserData - this class stores its own array of CUserProfiles (example class) and returns the relevant text or image in GetItemText andGetItemImage. The **only** mandatory functions required in a derived class are:

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

int GetItemCount()

**Description:**

Returns the number of items in the data source.

http://65.39.148.34/images/minus.gif Collapse | [Copy Code](http://65.39.148.34/Articles/13383/ListCtrl-A-WTL-list-control-with-Windows-Vista-sty?rp=/KB/wtl/WTL_ListCtrl/ListCtrlDemo.zip)

CString GetItemText(

int nItem,

int nSubItem

)

**Description:**

Returns the subitem text.

**Parameters:**

* nItem - Index to item (zero index).
* nSubItem - Index to subitem (zero index).

**History**

* 1.0 - (10th March 2006) First release.
* 1.1 - (13th March 2006) Small bug fix for VS2005 - thanks chengang.
* 1.2 - (17th March 2006) Corrected "Access violation" when compiled under VS2005, and added ShowThemed.
* 1.3 - (27th March 2006) Added auto-scrolling when group selecting - thanks for the suggestion, Slaham.
* 1.4 - (5th April 2006) Added smooth scrolling (SetSmoothScroll) and fixed Windows 2000 issues - thanks ihanz123.
* 1.5 - (10th April 2006) Added background image support and SetItemComboIndex - thanks for the suggestions, MyRock.

**Finally**

As wonderful as this list control is, I'm afraid I can't take all the credit, since it utilises all sorts of code garnered from all manners of sources, including articles posted on CodeProject. As you might expect, ideas and code snippets have been stolen and mercilessly twisted into this project, and in the process have adopted my coding style. Apologies to anyone who recognises their work and has not been credited (drop me an e-mail and I'll amend my acknowledgements), but to the following I give thanks for pointing me in the right direction.

Bottom of Form