Introduction

I have seen many requests in the Turbo Vision forum for a Combo Box class similar to the one available in Microsoft Windows. Here is a family of classes that provide the functionality of the Windows Combo Box.

Combo Box Classes

There are three classes that make up the Combo Box family: TComboBox, TComboWindow, and TComboViewer. TComboBox provides the icon in the dialog box that represents the whole TComboBox; by clicking on the icon or pressing the down arrow while the linked TInputLine is selected, the Combo Box selection window becomes active. The selection window is made up of a TComboWindow, a TComboViewer, and a TScrollBar. While active, the TComboWindow is the modal view; clicking outside of the window area causes the TComboWindow to terminate with a cmCancel command.

The TComboBox is only one character wide, so it should be place directly after the inputline to which it will be linked. For example:

```
  tv = new TInputLine(TRect(2,2,20,3), 25);
  insert(tv);
  insert(new TComboBox(TRect(20,2,21,3), tv, list));
```

The TComboWindow automatically sizes itself to be as wide as the inputline and the combo box together.

When used with a regular TInputLine, TComboBox acts just like a THistory object. However, when used with the TStaticInputLine (also in this zip file), TComboBox acts like a Combo Box in Windows, only allowing those items already in the list to be selected.

Class Descriptions

TComboBox

TComboBox implements a pick listof items for the user to choose. When used with a standard TInputLine, it acts very much like a THistory object. If used with a TStaticInputLine, only the items which are currently in the list can be selected.

Data Members
char *icon;
>	Points to the character that is used by the draw() function to display the icon on the screen.

TInputLine *link;
>	Points to the related TInputLine, or a descendant of TInputLine. Used to update the TInputLine data member when an item is selected from the list.

TCollection *list;
>	Points to the list of items that are currently available for selection by the user. This list is updated whenever the selected state of link is changed.

Member Functions
TComboBox(TRect& bounds, TInputLine *aLink, TCollection *aList);
>	Constructs a new TComboBox. If the extent of bounds is greater than a 1 by 1 square, the extent is changed to that size.

virtual ushort dataSize();
>	Returns the size of the data read or written by getData and setData. Currently returns sizeof(void *).

virtual void draw();
>	Draws the character pointed to by icon in the specified color.

virtual void getData(void *rec);
>	Returns a pointer to list in rec.

TPalette& getPalette() const;
>	Returns the defined palette cpComboBox.

virtual void handleEvent(TEvent& event);
>	Handles mouse events and a down arrow key event to open a TComboWindow.

virtual void newList(TCollection *aList);
>	Sets a new list for TComboBox by first deleting the old list and then setting list to aList.

virtual void setData(void *rec);
    Reads a pointer from rec and assigns it to list.

void shutDown();
    Sets link = 0 and list = 0, then calls TView::shutDown(). Deleting the actual
    TCollection is the responsibility of the program; this provides for a persistent list
    between calls to the same dialog box.

TComboViewer

TComboViewer is a descendant of TListViewer; it provides the actual list viewing mechanism for the TComboBox.

Data Members
TCollection *list;
    Points to the list of items that are currently available for selection by the user. This list is updated whenever the selected state of link is changed.

Member Functions
TComboViewer(const TRect& bounds, TCollection *aList, TScrollBar *sb);
    Constructs a TComboViewer by setting list = aList and calling TListViewer(bounds, 1, 0, sb).

virtual ushort dataSize();
    Returns the size of the data read and written by getData and setData.

TPalette& getPalette() const;
    Returns the defined palette cpComboViewer.

virtual void getData(void *data);
    Accepts data in the format of a TListBoxRec.

virtual void getText(char *dest, short item, short maxLen);
    Returns the itemth item from list in dest, not copying more than maxLen characters.

virtual void handleEvent(TEvent& event);
    Checks for keyboard or mouse events that will end the modal view of TComboViewer. All other events are passed on to TListViewer::handleEvent.

virtual void newList(TCollection *aList);
    Accepts a new list to be displayed, and disposes of the old list.

virtual void setData(void *data);
    Returns data in the format of a TListBoxRec.

void shutDown();
    Sets list = 0, then calls TListViewer::shutDown(). Disposing of the list is the responsibility of the programmer.

TComboWindow

Data Members
TComboViewer *viewer;
      A pointer to the TComboViewer that is a subview of TComboWindow.

Member Functions
TComboWindow(const TRect& bounds, TCollection *aList);
      Calls TWindow(bounds, 0, 0), the creates a TComboViewer and a TScrollBar to fit
      within its bounds. The aList parameter is passed on to TComboViewer.

TPalette& getPalette() const;
      Returns the defined palette cpComboWindow.

void getSelection(char *dest);
      Returns in dest the item selected from the list.

virtual void handleEvent(TEvent&);
      Checks for mouse events outside of the bounds of TComboWindow; all other
      events are passed on to TWindow::handleEvent.

void setSelection(char *data);
      Sets the selected item in the list to data.

TStaticInputLine

Data Members
TCollection *list;
    Points to the list of items that are currently available for selection by the user. This
    list is updated whenever the selected state of link is changed.

Member Functions
TStaticInputLine(const TRect& bounds, int maxLen, TCollection *aList);
    Created a TStaticInputLine by setting list = aList, and calling TInputLine(bounds,
    maxLen).

virtual void getNextMatch();
    A replacement for the call to matchFirstChar(). Currently getNextMatch calls
    matchFirstChar(), but later versions will have an algorithm to provide a circular
    search queue making TStaticInputLine behave more like Windows.

virtual void handleEvent(TEvent& event);
    Handles most keyboard events for TStaticInputLine; keyboard events handled
    include all printable characters, up and down arrow keys. All other events are
    passed on to TInputLine::handleEvent.

virtual void newList(TCollection *aList);
    Accepts a new list to be displayed, and disposes of the old list.

Boolean matchFirstChar(void *, void *);
    matchFirstChar is not a member or a friend function of TStaticInputLine, but is
    called by TStaticInputLine::handleEvent. This function is of type ccTestFunc and is
    used to test if the first characters of the two items passed are the same. This
    function is case insensitive.