Asgard ToolBox and **Asgard Tabs**

Asgard ToolBox and Asgard Tabs, background:

Like most software developers I wanted to have an attractive interface in my applications, using the latest 'toolbar' look, and later the tabbed dialogue boxes seen in various Microsoft applications. However I was not to keen on having yet another VBX to distribute and update, so I tried to figure out a way to do it in Visual Basic code, with a little help from the Windows API. It proved perfectly possible, and showed that VB is capable of anything you can do in C or any other language. (The VBX capability seem to have made us all a little lazy <g>).

The resulting code has now been used in several of my applications, and I am now releasing the source code for sale. This demo applications shows all the capabilities of the Asgard ToolBox and Asgard Tabs routines. If you would like to order the source code, go to How to Order and select print from the file menu.

Arild Bergh

Oxford, 9/12/93

Asgard Toolbox and Asgard Tabs, main features:

- Written completely in Visual Basic (with a few API calls)
This means:

- No VBX to distribute and update.

- Easily portable to other versions of VB (for NT and Mac for instance).

- Portable between your different projects. Asgard ToolBox and Asgard tabs is written in such a way that you only need to add a .BAS file (or two if you want to use both Tabs and Toolbox) to your projects. In addition you add a .FRM for the ToolBox.

- You can easily change the default behaviour of the toolbox or tabs by changing the Visual Basic code. For instance if you want pink tabs with orange polkadot main body, you can just change the code which actually displays that part of the tabs.

- Works under Visual Basic version 2 and 3

- Supports 3 types of toolboxes
You can have 3 types of toolboxes (and you can switch between any of them at runtime!): A 'toolbar' which are placed under the standard menus (if you use menus), a floating toolpalette similar to the one VB is using and a floating resizable palette similar to the one used in Lotus Applications.

- Predefined template to respond to user actions
Both Asgard Tabs and Asgard Toolbox issues a call to a sub routine where you handle the button/tab clicks. This is similar to the native Visual Basic way of working. For the tabs the subroutine is **Tab_Clicked** and for the toolbar it is called **Button_Clicked.**

- Multistate buttons
Asgard ToolBox supports 3 states for the buttons in the toolbox: UP, DOWN and DISABLED.

- Group of buttons
You can have a group of buttons, i.e. only one of the buttons can be down at a time. this is similar to the Visual Basic option buttons.

- Buttons which stay down or not
You can define whether a button should stay down once it is pressed, or if it should come out like the standard Visual Basic command buttons.

- Multiple toolsets
The toolbox supports multiple 'toolsets', i.e. you can change the type of tools which is displayed (with ONE single call to a sub routine) depending on the task at hand.

- Button properties changeable from within your code
As well as automatically handling the users clicking on a buttons, and changing it's appearance, you can change these from within your code: Disable, enable or set the UP/DOWN position of a button. Useful to give visual feedback to the user.

- Repeated clicks when keep mousebutton down
Asgard Toolbox also supports a 'repeater' effect: When a mouse button is pressed down, it repeats the calls to your code until it is let up again. This is similar to the Spin button in VB.

- Drag and drop support
As well as allowing you to click a button, you can also drag and drop controls onto the buttons.

- Switch between toolbar fixed under menu & floating tool palette
The toolbox can be switched between a floating toolpalette and a fixed toolbar (similar to Lotus programmes). This gives you flexibility to change it depending on your type of application (or user preference).

- Freezing a button
You can temporarily 'freeze' a button. this means that the user can not click the button, and no call will be generated to your code.

- Support for statusbar info

Both the Tabs and the Toolbox calls a sub routine in your code with the number of the tab or button as the user moves the mouse over it. This means that you can inform your users what a particular button is used for when the mouse passes over it.

- Move mouse away support

The toolbox provides proper 'move away' support, i.e. if the user clicks a button, but then moves the mouse pointer away, the button will immediately come out again, just like the standard windows buttons.

- Tabs both on top or bottom of a tab set

Asgard Tabs supports the use of tabs on the top as well as the bottom of a tab set (although not on the same set). This allows you to (for instance) emulate a time management planner, with tabs on both sides.

- Use any font in any size

You can use any font you want to for the tabs, in any size and colour. You can also define the size of the actual tab in each separate tabset.

- Adding and deleting tab sets at runtime

You can add and remove any number of tabs from within your code at runtime. Just one line of code is required.

- Enable/Disable tabs

Any tab can be enabled or disabled at runtime, for instance in order to lock an option in response to what the user does.

- Querying the settings

Both Asgard Tabs and Asgard ToolBox allows you to find out which tab is the selected one, or what the state of a particular button is. This means that you do not need to keep track of the individual values yourself, but can query the Toolbox/Tabs functions.

The basics of Asgard ToolBox and Asgard Tabs

To create a toolbox or a tabset you simply call a Visual Basic routine with parameters describing how you want the tab or toolbox to look: For instance the font to use on the tabs or the number of buttons in a toolbox.

To react to clicks you simply provide code in two subroutines: Tab_Clicked and Button_Clicked. To give info to the user about the use of a button or a tab you provide code in two subs called Tab_Info and Button_Info

To initialize a tabset, you call Tab_Add, and to initialize a toolset you call Set_Toolset. In addition, for the toolbox, you need to create a single bitmap for each toolset (not for each button!).

Asgard Toolbox/Tabs also provides functions which let you query the state of a toolbutton or tab: GetButtonInfo and GetTabInfo.

Thats all folks; as simple as that.

HOW TO ORDER ASGARD TOOLBOX/TABS SOURCE CODE:

To order via CompuServe:

GO SWREG at any ! prompt. This will take you to the ShareWare registration database. Follow the onscreen instructions on screen. The programme ID for Asgard Toolbox/Tabs is 2017. The cost is 22 US dollar.

To order via Mail: (Cheques and Money orders only)

Select print from the file menu to print out this order form and fill in the required fields.

Please send me Asgard Toolbox/Tabs for Visual Basic:

Name: ____

Address: ____

____

____

Country: ____

Tel: ____

EMail: ____

Where did you fint Asgard Toolbox/Tabs: ____

Send by: Mail__ E-Mail__

Cheque/money order for 22 US$ enclosed. If you live outside the UK add 2 US$ for postage and packaging!

What you receive:

When you purchase Asgard Toolbox/Tabs you will receive the full source code which is well documented as well as a Windows helpfile documenting all the functions you need to call, or which will be called from Toolbox/Tabs. You also receive the source code of this application.

This will be emailed to you if you order via CompuServe or mailed if you specify so on the mail order form above.

We will also provide help to any problems/questions you have with the code via E-Mail (CIS or CIX).

Asgard Software
87 Lytton Rd
Oxford
OX4 3NY
UK

Tel: +44 (0)865 772160
Fax: +44 (0)865 771461
EMAIL: CIS: 100113,431
aberji@cix.compulink.co.uk