



# ClassAssist.<sup>TM</sup> Class Manager for Microsoft<sup>®</sup> Visual Basic<sup>®</sup>.

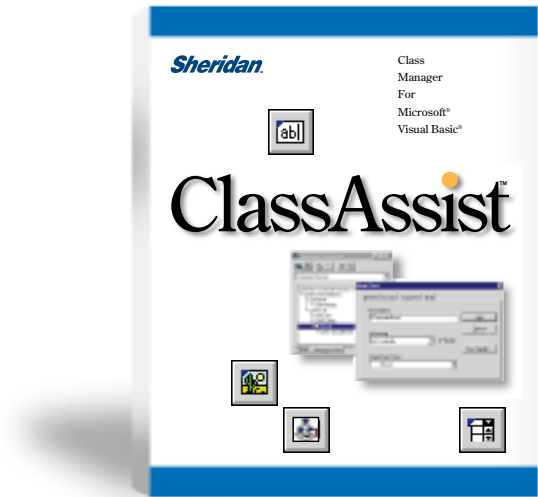
ClassAssist is a revolutionary new productivity tool from Sheridan that lets you exploit the full capabilities of classes in Microsoft Visual Basic 4.0. It adds features and capabilities that make Visual Basic classes more OOP-like, more powerful and easier to use. ClassAssist consists of 3 major elements:

- An intuitive IDE (Integrated Development Environment) for creating and managing hierarchies of reusable classes. The IDE also supports team development with integrated check-in and checkout facilities.
- A set of 6 visual base classes from which you can derive new classes that let you create specialized custom controls using Visual Basic code.
- Sheridan WinAPI Oblets<sup>™</sup> – programmable objects that simplify access to the Microsoft Windows<sup>®</sup> API when developing your classes. Oblets extend Visual Basic syntax by becoming part of the language much like the intrinsic Visual Basic database objects.

ClassAssist leverages the raw power of Visual Basic classes by adding inheritance, property and method overrides in derived classes, better encapsulation and method overloading. ClassAssist's ability to create reusable classes that inherit functionality from existing classes is a major step forward in productivity for Visual Basic developers. Now you can experience the benefits of code reuse just as C++ programmers have been doing for years. And since ClassAssist supports team development with its integrated checkin and checkout facilities, these benefits can be realized by the entire group.

## The ClassAssist IDE

With ClassAssist, creating reusable classes that inherit functionality from existing classes is as simple as pointing and clicking. And overriding inherited properties or methods is just as easy – select the item to override and click!



The Class Explorer (ClassAssist's main window) shows the relationships between classes you've created in a clear, easy-to-read outline view. To use a class in a project, simply drag and drop the class from the Explorer to your VB project window (or to a VB designtime form for visual classes).

Class hierarchies can be organized into separate libraries which can be designated either private or shared. And since the ClassAssist IDE provides integrated checkin and checkout facilities, classes in public libraries can be managed by teams of developers without worrying about overwritten code.

## Creating Custom Controls Using ClassAssist and Visual Basic Code

ClassAssist lets you create specialized OCX controls for use in your Visual Basic projects by deriving classes that inherit functionality from any of the five visual base classes supplied with the product. To create a new control simply derive a new class from one of those provided, then override selected properties and methods to implement the behavior and appearance you need. And you do it all using Visual Basic code!

## Sheridan WinAPI Oblets

WinAPI Oblets simplify your usage of the Windows API by eliminating the need for function and constant declarations and letting you use familiar Visual Basic Object syntax when calling API functions. WinAPI Oblets gives you object-like access to all commonly used API functions which makes your code cleaner and easier to read. WinAPI Oblets also enables you to achieve more complete encapsulation of your classes. For example, the timer Oblet provides the same functionality as the Visual Basic Timer control, but since it doesn't require a form in order to be created, you can totally encapsulate timer functionality in any class without having to rely on an external form module being present.

# ClassAssist™



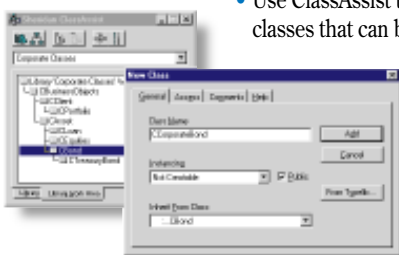
- The heart of the ClassAssist IDE is the Class Explorer. It shows the relationships between all classes in the current library in an easy to read outline view
- Right-click the mouse in the Explorer to derive a new class, edit properties and methods of an existing class or checkin or checkout classes in a shared library



- The ClassAssist code window lets you edit declarations, properties and methods in a structured manner

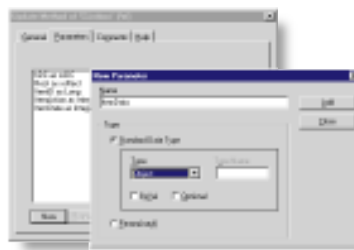


- ClassAssist lets you choose between variable-type properties and procedure type properties (Get/Let/Set)
- For procedure type properties, ClassAssist can optionally generate a private variable for you to store the property's value



- Use ClassAssist to create libraries of reusable classes that can be shared by all developers in your group. Create visual classes like specialized listboxes, or make non-visual classes that encapsulate important business rules

- ClassAssist even lets you define a property as a static variable which means that only one instance of the property is created but it's available to all instances of the class. This lets you totally encapsulate, within a class, any data that must be shared by all instances of the class without duplicating the data or relying on an external module to hold the data



- ClassAssist walks you through the process of defining methods for your classes. You can even define overloaded methods - i.e., methods with the same name that take different sets of parameters



- Visual Classes are derived from one of 6 supplied base classes. Listbox, Combobox, Command Button, State Button, Textbox and Canvas. By creating a new class that is derived from one of these 6 classes, you can create specialized custom controls to use in your project

- For those special situations where you need to process specific Windows messages, ClassAssist

lets you define a class message map which causes messages of the chosen types to be routed to a method in your class for processing



- Integrated checkin and checkout facilities mean that you can share libraries of classes between team members with confidence - a class can only be updated by one developer at a time

## Sheridan's state-of-the-art *WinAPI Oblets* puts the Windows API at Your Fingertips!

For example, to use a timer Oblet from within a class:

```
Dim MyTimer as New ssTimer
MyTimer.Interval = 250
MyTimer.Connect Me, "Alarm"
MyTimer.Enabled = True
```

'milliseconds  
'calls the 'Alarm' method of your class  
'when the timer expires  
'starts the timer

With WinAPI Oblets you never have to declare API functions or constants. Simply dimension the Oblet that contains the API functions you want to use, and you're ready to access methods and properties immediately! ClassAssist lets you take advantage of Sheridan's advanced Oblet technology today!

Oblets include ssTimer, ssCommonDialog, ssHook (to install hooks like Mouse or Keyboard), ssMenu, ssWnd (with subclassing feature), ssBrush, ssPen, ssRect, ssEnumerator (for accessing APIs that require callbacks), ssSystem (for getting system info like System Metrics or System Colors) and a whole lot more!!

## Sheridan™

Reusable Components and Productivity Tools  
for the Visual Developer

**Sheridan Software Systems, Inc.**

35 Pinelawn Road, Suite 206E, Melville, NY 11747

Voice: (516) 753-0985 • Fax: (516) 753-3661 • BBS: (516) 753-5452

CIS: GO SHERIDAN • <http://www.shersoft.com>