CHAPTER 20

What Makes A Great Application for Windows 95?

While Windows 95 provides compatibility for running MS-DOS and Win16–based applications, users benefit from additional functionality supported by Win32–based applications. Win32–based applications benefit from the preemptive multitasking architecture of Windows 95 and the increased robustness and protection for running applications. In addition, there are six key areas that make a great application for Windows 95 from the users perspective:

- The Win32 Application Programming Interface (API)
- OLE functionality
- Windows User Interface Style Guideline 4.0
- Support for handling Plug and Play events
- Support for manipulating long filenames
- Adherence to common setup guidelines for consistent software installation

In the next section discusses why these components make these applications great for users.

The Win32 Application Programming Interface

Microsoft supports the use of the Win32 API on three operating system platforms: Windows NT, Windows 95, and Windows 3.1 with Win32s. Each operating system supports a common set of Win32 APIs, allowing applications to be developed for a single API set and run on multiple platforms. This allows application developers and corporate developers to learn a single API set, and leverage development resources to support a broad base of hardware systems. Users benefit from being able to run the same application on multiple platforms, and increased system reliability under Windows 95 due to improved robustness and memory protection available to 32-bit applications.

Windows 95 delivers a robust and powerful 32-bit platform. 32-bit applications for Windows 95 are preemptively multitasked, run in private address spaces, and can spawn multiple threads of execution. Preemptive multitasking ensures excellent system responsiveness, allowing users to run multiple applications simultaneously and integrate personal productivity and business-critical applications in a smooth manner. This is similar to the model that Windows NT uses today. The use of a private address space for each Win32–based application ensures that multiple applications can run simultaneously without interfering with each other or the operating system itself. The Windows 95 operating system provides smooth, preemptive multitasking and protected virtual memory, because Windows 95 is based on a re-architected 32-bit protected-mode kernel and a 32-bit protect-mode driver model.

User Benefits of Using Win32-based Applications

Running 32-bit applications under Windows 95 provides the following improvements from an end-user perspective:

- Background multitasking for running multiple applications that is smoother because of the preemptive multitasking architecture in Windows 95
- Overall system performance improvements due to 32-bit operating system components
- Robustness and system reliability improvements due to 32-bit memory protection and separate message queues
- New applications functionality because of Win32 and other operating system services
- File manipulation that is easier because of long file name support in Windows 95

OLE Functionality

Users are getting and using more applications per PC than ever before. In 1992, InfoCorp reported the average number of applications purchased per desktop running the Windows operating system increased to more than 7 programs, up from an average of 3.4 programs for customers using the MS-DOS® operating system in 1986. People are not just acquiring more applications, they are using them together. Research shows that users cite the ability to move and share information among applications as the most important reason for using windows applications.

Users who learn one Windows application find it easy to learn a second or third. So, as users access several applications in the course of creating a compound document, they'll feel comfortable with those applications.

The Solution for Application Integration

OLE is a mechanism that allows applications to interoperate more effectively, thereby allowing users to work more productively. Users of OLE applications create and manage compound documents. These are documents which seamlessly incorporate data, or objects, of different formats. Sound clips, spreadsheets, text, bitmaps are some examples of objects commonly found in compound documents. Each object is created and maintained by its server application. But through the use of OLE, the services of the different server applications are integrated. Users feel as if a single application, with all the functionality of each of the server applications, is being used. Users of OLE-enabled applications don't need to be concerned with managing and switching between the various server applications; they focus solely on the compound document and the task being performed using OLE-based features.

Features of OLE

With OLE, Microsoft Windows 95 increases the degree of application integration available to any applications which take advantage of the services. This gives users tangible benefits, allowing them to share data and functionality across applications, and to combine them as they please. Because OLE is based on an open industry-standard, users can extend their applications with additional third-party products, further expanding their choice and flexibility.

OLE provides the following features to allow users to easily integrate information into multiple applications:

- **Cross-application drag-and-drop**

  Users can drag-and-drop graphs, tables, and pictures directly onto slides, worksheets, and documents to mix text, data, and graphics into compound documents. Using drag-and-drop is faster and more intuitive than using the Clipboard to cut and paste.

- **Visual editing**

  With visual editing, the user can double-click an object to directly edit it while remaining in the original document. For instance, double-click an embedded Microsoft Excel worksheet in a Word document, and the Microsoft Excel menus and toolbars automatically appear within the context of Word. Unlike the first release of OLE technology, the user is not launched into a separate Microsoft Excel window to work on the spreadsheet data.

Drag-and-Drop

Drag-and-drop is a new and more intuitive way to move data between applications. The most widely used way to transfer data between applications has been to use the Clipboard. But this involves multiple steps; namely using the Copy operation, moving to the destination application, and using the Paste command. A more effective way to move information—drag-and-drop—already exists within applications and, with OLE, it now works between applications, too. The user simply selects an object in one application, drags it to its destination in another application and drops it into place. Objects also can be dragged over the desktop to system resource icons such as printers and mailboxes, making it faster and easier to send, print, or share files.

Visual Editing

Visual editing makes revising a compound document faster, easier, and more intuitive. For example, a Microsoft Excel worksheet that’s contained within a Word document can be double-clicked by the user. The user then is able to interact with the Microsoft Excel worksheet right there, without switching to a different application or window.

The menus and toolbars necessary to interact with the Microsoft Excel spreadsheet temporarily replace the existing menus and controls of Word. Microsoft Excel, the application that is needed to edit or modify the spreadsheet, partially “takes over” the Word document window (see Figure 1). When the user wants to work on the word processing portion of the document, the focus returns to Word, and the original Word menus and controls are restored.

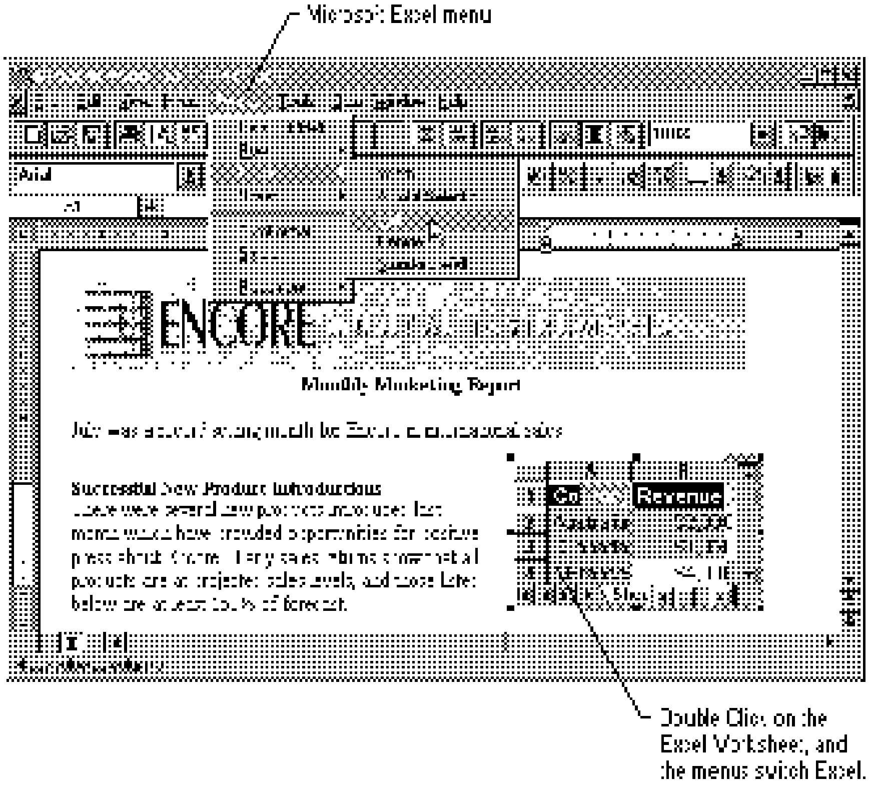
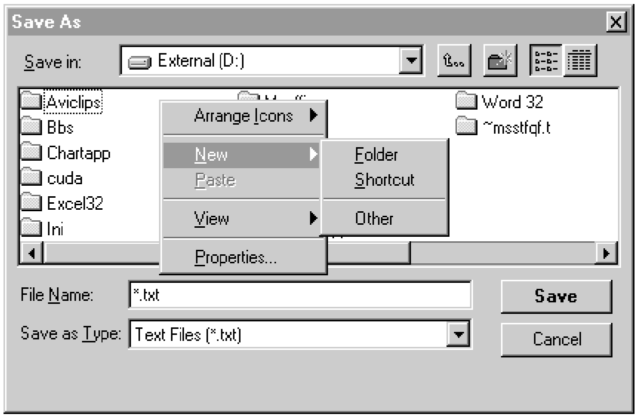

Microsoft Excel menu
ENCORE
Monthly Marketing Report
Successful New Product Introductions
Revenue
Double Click on the Excel Worksheet, and the menus switch Excel.

Figure 1. Activating OLE Object Changes Menu Context

The advantage of visual editing is even greater for example, when users create compound Word documents including large numbers of objects created by different applications, such as Microsoft Excel worksheets and charts, PowerPoint graphics, sound and video clips, and so on. Instead of switching back and forth among different windows to update the objects, the user is presented with a single document window in Word, providing a single location for editing and other interactions with the data. Visual editing offers users a more “document-centric” approach, putting the primary focus on the creation and manipulation of information rather than on the operation of the environment and its applications.

Windows User Interface Style Guideline 4.0

As in previous version of Windows, one of the reasons why applications are easy to learn is the fact that they look and act alike. With Windows 95, Microsoft has taken great steps to improve the basic common controls that all applications can share. These controls have evolved based on user feedback and extensive usability testing here at Microsoft. Applications that use these controls provide the users with commonality and improved features—such as being able to create new folders in the Save As dialog box—without having to switch to the Windows Explorer or File Manager.



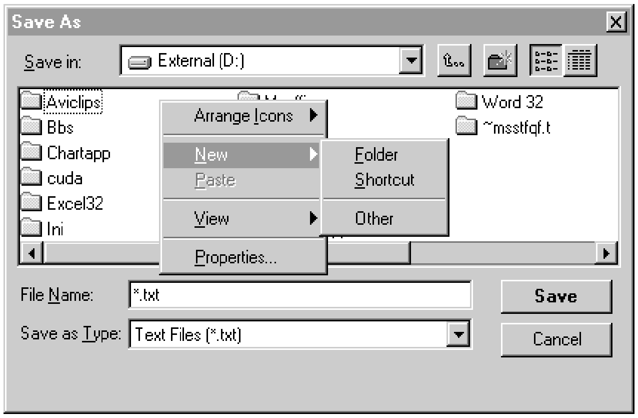
Save in: External (D:)

File Name: *.txt

Save as Type: Text Files (*.txt)

Figure 2. Save As Common Dialog

In the new Printer Properties dialog you can see examples of some of the new controls with make access to features even easier for users. At the top of the dialog you can see tabs for “Paper,” “Device Options,” Graphics,” and, in the case of the printer shown in Figure 3, “PostScript.” Clicking any tab presents the user with properties for that particular area. Another new control available to all applications is the Spin Control next to the number of Copies.



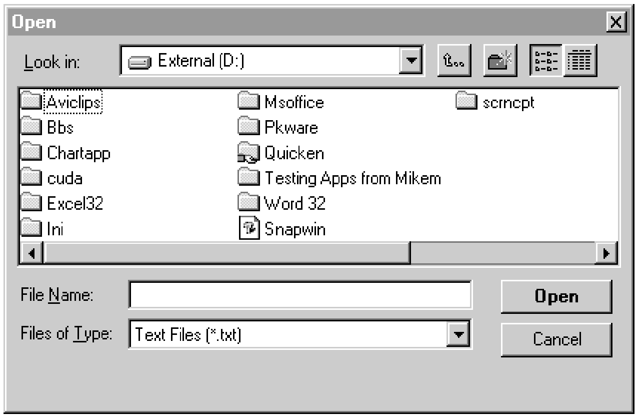
Save in: External (D:)

File Name: *.txt

Save as Type: Text Files (*.txt)

Figure 3. Sample Tabbed Dialog Box Property Sheet

The new Open Dialog allows the user to see long filenames, and navigate the entire PC, and connected network to look for files to open.



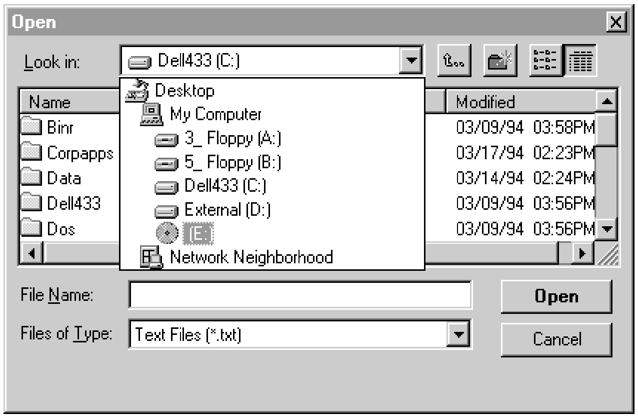
Look in: External (D:)

File Name:

Files of Type: Text Files (*.txt)

Figure 4. Open Common Dialog

The new Open dialog also uses Tree Lists to allow the user to navigate through the hierarchy of the hard disks attached to the computer and through the network to which the computer is connected.



Look in: Dell433 (C:)

Name	Modified
Binr	03/09/94 03:58PM
Corpapps	03/17/94 02:23PM
Data	03/14/94 02:24PM
Dell433	03/09/94 03:56PM
Dos	03/09/94 03:56PM

File Name:

Files of Type: Text Files (*.txt)

Figure 5. Open Common Dialog Provides Easy Access to Network Resources

Figure 6 shows another new control that makes viewing and accessing hierarchical information even easier. This is a tree list control found in the property sheet for the Device Manager in the System section of Control Panel. As users expand and collapse the tree, they can see information relevant to their topic of choice.

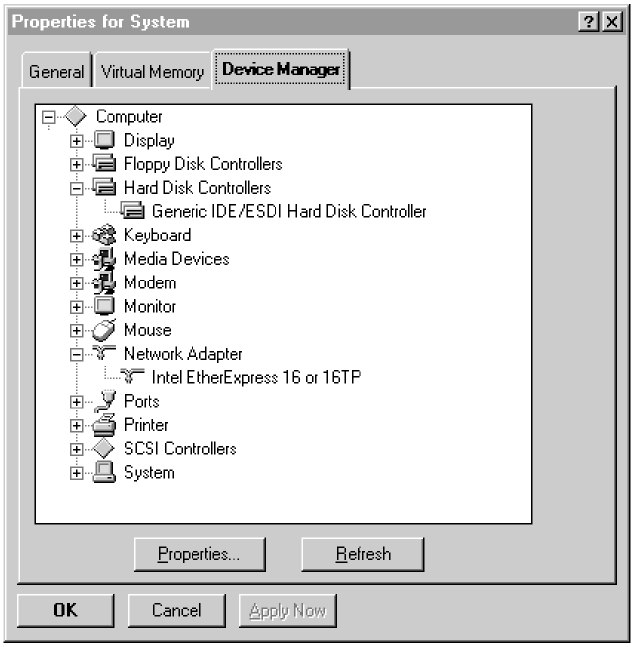


Figure 6. Sample Tree List Control

Applications no longer have to include their own custom slider controls. Figure 7 shows the new common slider control included in Windows 95.

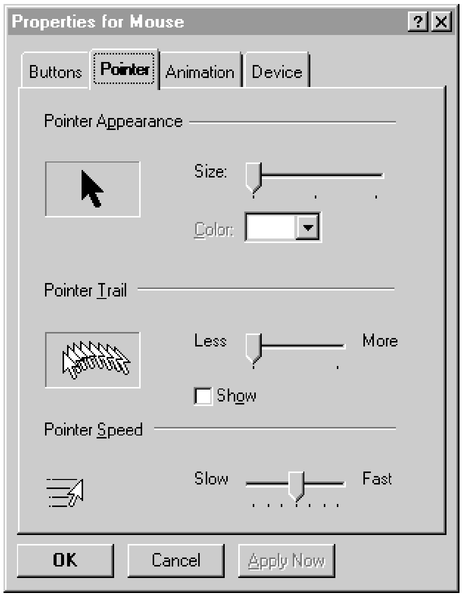


Figure 7. Sample Slider Control

There are many new common controls, tool bar, status bar, column heading, tabs, slider, progress indicator, rich text control, list view, tree view... and much more. Great applications for Windows 95 will use these new controls to make the users access consistent across applications and make the entire system much easier to use.

Support for Plug and Play Events

Applications that provide Plug and Play event awareness help users by seamlessly adapting to hardware configuration changes. Users will reap two key benefits:

u **Applications automatically recognize and respond to hardware changes**

Consider this scenario: the desktop Control Panel in Windows 95 recognizes changes in the video resolution and configures the computer accordingly. Now suppose a user who has a mobile PC installed in a docking station and who is using an external monitor running in 1024 x 768 resolution mode. When the user undocks the PC, the desktop Control Panel recognizes this action and seamlessly switches resolution for the mobile PC to 640 x 480. When this change occurs, Plug and Play aware applications will resize their windows and toolbars accordingly. The user doesn’t have to do a thing; it’s all automatic.

Here's another scenario: Suppose you were using a mobile PC to work on a document. One of the most critical situations you could find yourself in is running out of battery power. With Windows 95, your computer sends a message to all the active Plug and Play aware applications, telling them to shut down and save your data because the power is going off in a few minutes.

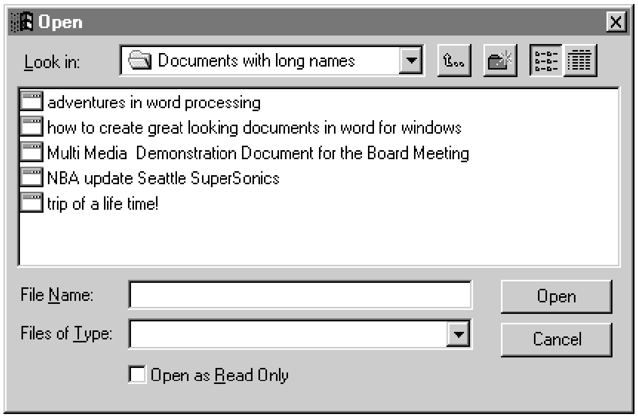
- **Applications warn users about open network files when hot-undocking their computers**

  Suppose that you have a PCMCIA network card installed in your laptop computer, and you are leaving the office. As you leave the office, you switch PCMCIA cards and install your modem for dial-up network access. With Plug and Play, you don't have to fuss with software configuration—you're set. Windows 95 simply knows that the network has gone away, and that the network card is now replaced by a modem. A Plug and Play aware Email application also learns this information from the operating system. It knows that there is no more network connection and it now needs to use the modem to make connections. The software configurations are automatically made for you.

It should be quite clear that applications which are Plug and Play aware provide seamless adaptation to changes in the hardware configuration. With applications that are Plug and Play aware, users can focus on their work, not their configuration.

Long Filename Support

As you have seen by now, there is a much improved new shell for Windows 95. But the shell itself is only part of what is really here for users. Now an application that takes advantage of this new shell support can offer their users long filenames and direct file viewing. Long filename support means that documents no longer have to be limited to eight characters for names. They now can have up to 255 characters. Instead of "23ISM_JB.doc," you can name a file "Status report July 23 regarding the ISM project for my boss Jim Bernstein"—a title that really tells you what the document is about. Applications that support the Viewer capabilities in Windows 95 provide users with a quick and easy way to view their documents directly from the shell without launching the application.



Look in: Documents with long names

adventures in word processing
how to create great looking documents in word for windows
Multi Media Demonstration Document for the Board Meeting
NBA update Seattle SuperSonics
trip of a life time!

Figure 8. Windows 95 Applications Support Long Filenames

Consistent Setup Guidelines

In the past users have generally had an easy way to setup their new applications, but removing these applications from their hard disks was not so simple. Most Windows 3.1 users eventually ended up with all kinds of files on their systems which were never used because they belonged to some previously-deleted application. And since many applications use the same library files, with Windows 3.1 it was quite common to have several copies of a files installed in different places on the hard disk, an inefficient use of precious disk space. Another common problem with Windows 3.1 is “files, files, files, everywhere”—applications put files in their own directory, in the Windows directory, in the System directory, in the root, you name it. This creates a real mess when trying to keep track of what’s where.

The *Software Development Kit for Windows 95* offers to developers some guidelines for consistent installation locations and uninstall functionality in their applications. Common libraries can be shared by applications thereby reducing the amount of disk space consumed by duplication. The guidelines also set standards for where developers should put all their files on the hard disk. No more “files, files everywhere” syndrome. These guidelines provide a much easier, powerful, and compatible structure for users. Setup programs that follow the guidelines will all operate similarly, use consistent naming conventions, and offer the same setup options—thus reducing the learning curve for users, and improved manageability and support for corporations allowing for increased efficiency in remote administration installed software.

BLANK PAGE

IMPORTANT: This text will appear on screen, but will not print on a PostScript printer.

This page should be the last one in this file; it was inserted by running the InsertBlankPage macro.

Do not type any additional text on this page!