

International Language Support

With the growth of the worldwide PC market, Microsoft Windows and Windows-based applications have made PCs easier to use around the globe. The fact that Windows and Windows-based applications are sold and used worldwide poses some unique problems for both Microsoft as an operating-system vendor and ISVs as application developers.

When a new software application or operating system intended for a world market is developed, efforts must be made to localize the software to the countries and written languages in which it will be used. In many cases, localization is a simple matter of translating the names of menus, menu items, and strings displayed by the software into the language used in the locale. However, as the features and functionality of a software product grow, so does the complexity required to tailor the application to characteristics of the native country. Since the start of the design work for the Windows NT operating system, Microsoft has been adding to the level of support for international languages and cultural conventions in the 32-bit editions of the Windows family of operating systems.

This section discusses the localization plans for Windows 95, the built-in international support for using Windows 95 on a worldwide basis, and the special provisions that Windows 95 includes for enhancing existing or developing new applications that can be used in different parts of the world.

Summary of Improvements over Windows 3.1

Support for using the Windows operating system on a global basis is improved in Windows 95. The benefits for both users and software developers are summarized below.

The benefits for users include the following:

- **Easy switching from one language to another.** Windows 95 makes both the use of multiple language fonts and character sets, and switching among the different keyboard layouts required to support them, easy.

Included in the CD-ROM version of Windows 95 (all languages) is an option that lets any application **write text** in not only Western European but also in all Eastern European Languages. (note that you can only **write text** in these languages, if you need to run for example a Russian version of Microsoft Excel you will need either Russian or the Pan European Edition of Windows 95). To install the Multilingual support, you either choose it when you setup Windows 95 or install it later through

Control Panel (Add/Remove programs icon, Windows Setup tab, Accessories, Multilingual support.)

With the old Eastern European version of Windows 3.1, users can directly switch between only two keyboard layouts—for example, Russian and English. With the standard Latin versions of Windows 3.1, users cannot easily switch between different keyboard layouts; they have to go to the Control Panel for each language switch. As an added complexity you can only switch between languages within the Latin hemisphere, you can't switch to for example Russian.

With Windows 95 this is now an easy and seamless operation, users can easily switch among all available languages and corresponding keyboard layouts configured on their system by using the ALT+SHIFT key combination or click on the language indicator on the taskbar, making the creation and integration of information in a multilingual document easy.

- **Font substitution.** Windows 95 substitutes fonts when switching among different languages if the original font is not present on the system. When switching among different languages, matching fonts for the new language are substituted if the original font is not available. As a result, users can read and use the text for a similar character set, even if they don't have the font that the original information was created in.
- **Correct sorting and formatting rules.** Different locales and cultures have different rules for interpreting information. For example, cultures use different sequence algorithms for sorting information, use different comparison algorithms for finding or searching for information, and use different formats for specifying time and date information. Win32-based applications that use the National Language Support (NLS) APIs allow users to easily exchange information on a global basis, while preserving the integrity of the information.

The benefits for developers include the following:

- **Easy addition of international language support to applications.** Developers can now use the Win32 NLS APIs for sorting, searching, and manipulating information in a locale-independent way. NLS services in Windows 95 ensure that information is handled correctly for the given culture or locale. The correct national format is automatically supplied based on the international settings specified by the user in the Control Panel. For example, to obtain the current date format information to match the current locale, the application calls an NLS API, and the system returns the correct format. Likewise, to sort information in the proper sequence in French, Norwegian, or Spanish, the application calls a corresponding culture-independent NLS API.
- **Automatic switching of fonts and keyboard layouts.** Windows 95 provides services that application developers can use to ensure that as users move through a multilingual document, the correct fonts and keyboard layouts are used. For users who create or edit multilingual document content—for example, translators—a Win32-based application that uses the international services in Windows 95 automatically activates the correct fonts and corresponding keyboard layouts for the edit point in the text. This feature allows easy editing of information contained within multilingual documents.
- **Preservation of language-specific attributes on the Clipboard.** Windows 95 provides additional services for application developers so that information can be

passed through the Clipboard to easily exchange information between internationally-aware applications, while preserving all language formatting characteristics.

- **Switching of languages by multilingual-aware applications.** Windows 95 provides services that application developers can use to automatically switch the language that the system uses to match attributes in a document. For example, as users scrolls through a multilingual document, the application can automatically switch the system language to match the format of the information contained within the document.
- **Storage of international language information in RTF format.** Extensions have been provided to the RTF specification to support saving language-relevant information in Rich Text Format (RTF) from a multilingual-aware application.

The Localization of Windows 95

As a result of the success of Microsoft Windows around the world, Windows and Windows-based applications have been localized into many different languages. Microsoft Windows 3.1 was localized into more than 25 major languages, a process that took as long as 18 months and delayed the availability of Windows 3.1 for some language versions. With Windows 95 international localization, it is much easier to do a quality localization in shorter time. To better support a global market, Microsoft plans to localize Windows 95 into at least 29 different language versions, including German, French, Spanish, Swedish, Dutch, Italian, Pan European Edition (an English version that supports running Eastern European applications), Norwegian, Danish, Finnish, Portuguese (Brazilian and Iberian), Japanese, Chinese, Korean, Russian, Czech, Polish, Hungarian, Turkish, Greek, Arabic, Basque, Hebrew, Thai, and Catalan (as well as several variations of these languages). The localized versions of Windows 95 will be released on a planned development schedule that does not exceed 120 days—exceptions being Arabic and Hebrew that will release in about 180 days after the English version due to the added complexity of bilingual switching (switching between English and for example Arabic menus).

International Language Issues

Localization is only a small part of the effort that goes into ensuring that an operating system can be used effectively in a worldwide environment. A worldwide operating system must also provide services to support the use of international applications and to support the global market by making the application developer's job easier. This section discusses some of the language issues that international users and application developers face:

- **From the user's perspective.** Some users need to include more than one language in a document. For example, they might be translating from English into Russian or they might be writing a product instruction manual in many different languages. When using more than one language, users must deal with a series of obstacles. For example, they must repeatedly switch to another keyboard layout on-the-fly so that they can continue writing in a different language. When using a database, users face the problem of sorting the information in the correct order for a given language.
- **From the developer's perspective.** When localizing a product into different languages, developers are faced with several questions, such as the following: "What

is the correct sorting order for French?” “How is a date represented in Germany?” “Do the Swedes really need to have the ability to use the characters Å, Ä, and Ö?” “If a document contains text in more than one language, is there some way for the software to know which part of the document is in which language?” “Can information in a multilingual document be passed to another application via the Clipboard?” Many developers try to address these issues in their applications and fall short, creating problems for the users, their support organization, and their own development team.

Because the mainstream Windows platform has not previously offered international language support as an operating system service, many application vendors have hard-coded global characteristics into their applications. Hard coding allows their applications to be used in a given locale, but prevents the applications from being used easily in a different cultural environment. As a result, users depend on application developers to provide a version of the application that matches their locale attributes.

In Windows 95, Microsoft has set out to offer international language support at the operating system and API level. This support adds functionality that provides solutions for using software and exchanging documents around the world. Providing international language support services in Windows 95 makes it easier for application developers to solve international language issues related to presenting or manipulating information in their applications. This section discusses those issues.

Date and Time Formats

Date and time information needs to be represented in different formats depending on the locale where the information is being used. For example, date information presented in American English places the day between the month and year, as in “March 9, 1994,” whereas a different locale may represent the same date as “9 March 1994.”

Sorting and Searching

International language issues are much more complex than simply representing date and time information in the correct format. Sorting and searching algorithms in applications must correspond to the proper language rules for the locale in which the information is being used and manipulated. The following examples illustrate the subtle differences between language rules:

- In French, diacritics are sorted right to left instead of left to right, as in English.
- In Norwegian, some extended characters follow the Z character because they are considered unique characters rather than characters with a diacritic.
- In Spanish, CH is a unique character between C and D, and Ñ is a unique character between N and O.

As a further example, if a database in Swedish is sorted with an English-language sort algorithm, the names would be sorted as shown in the left column of this table:

How Are Names Sorted?	
English sorting	Correct Swedish sorting
Andersson	Andersson

Åkesson	Karlsson
Ärlingmark	Magnusson
Karlsson	Turesson
Magnusson	Åkesson
Turesson	Ärlingmark

The system treats the Å and Ä as an A and therefore sorts them after A at the top of the list. However, in the correct Swedish sort order, the Å and Ä are sorted after Z because they are separate vowels that occur at the very end of the alphabet. A Swede looking for “Ärlingmark” would be confused to find it near the beginning, instead of at the end, of a list of names.

With Windows 3.1, many developers came up with their own sorting routines for different languages and hard-coded this functionality into their applications. Their applications are too inflexible to support the numerous right sorting tables required for all the languages into which they might want to localize their applications.

National Character Sets, Keyboards, and Fonts

In standard Windows 3.1, fonts native to the Eastern European countries, such as Greece, Russia, and Turkey, cannot be used. For example, if users tried to install a Russian font with an English or French version of Windows 3.1, the characters appeared unintelligible on the screen, and users couldn't use the font. To solve this problem, a special English Eastern European version of Windows 3.1 was designed for English users who needed to use Eastern European fonts, including Russian Cyrillic or Greek. The English Eastern European version of Windows 3.1 offered the same capabilities as the true Russian or Turkish Eastern European version of Windows for displaying font and character information.

The Solution: Multilingual Content Support

Windows 95 resolves many of the problems related to international language issues by integrating multilingual content support in the core of the operating system. Windows 95 also offers national language support to application developers as a series of APIs that are part of the Win32 API set.

Multilingual content support is the ability to display and edit text of various languages and scripts in a single document. Multilingual content support is a core feature of Windows 95 and will be also be provided in the next major release of Windows NT (code-named Cairo).

Multilingual content support in an application provides the following two major benefits:

- Users can create and edit documents with content in multiple languages and scripts and exchange these documents with users of other language systems. This feature is important within the European Union, for example, where Greek and Latin-based languages must coexist in documents.
- An application that supports multilingual content supports the native content of any market into which it is sold.

Easy Switching Among Languages and Keyboards

Windows 95 allows users to add support for multiple keyboard layouts to match different international conventions. In the Control Panel, the Keyboard icon provides the ability to configure the system to support the preferred keyboard layouts, as shown in Figure 115.

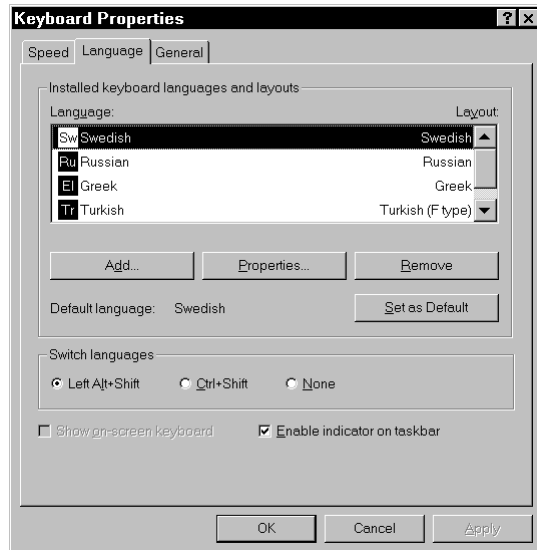


Figure 115. The Keyboard property sheet, showing international layout support

To change the keyboard layout in Windows 3.1, users had to go to the Control Panel each time they wanted to switch to a different keyboard layout. In Windows 95, switching keyboard layouts is much easier. Figure 116 shows a sample legacy word processing document that illustrates the ability to integrate text by using the Lucida Console font in different languages within the same document. The language identifier in the status area of the taskbar allows users to easily switch the system language among the available language options. A Windows 95 application that uses NLS APIs would incorporate the ability to switch the preferred language directly on the Toolbar of the application.

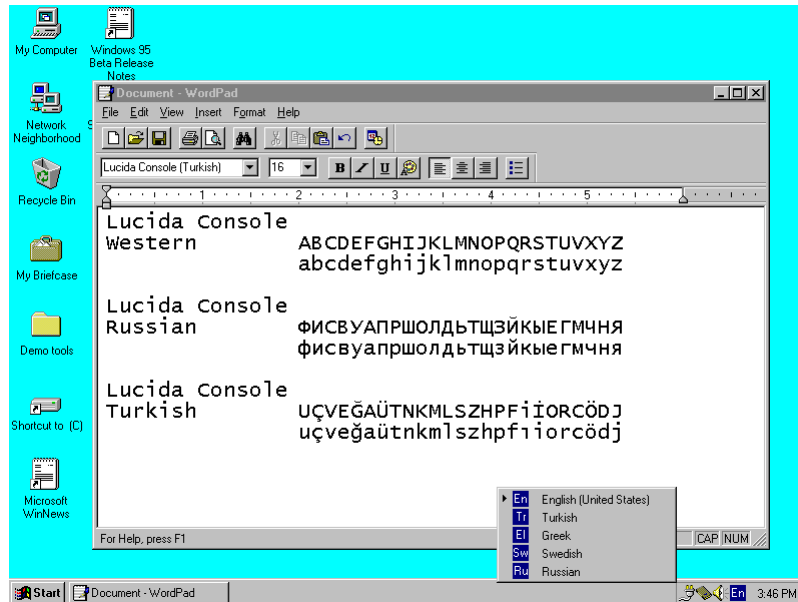


Figure 116. Switching among different languages to create a multilingual document

Multilingual Extensions to the ChooseFont Dialog Box

The ChooseFont common dialog box has been enhanced to include a list box showing the character set scripts supported by a particular font. This mechanism ensures the correct representation of fonts for a given language.

Figure 117 shows the new ChooseFont common dialog box, illustrating the integration of font script selection options. The Font Script list shows the script names for each of the character sets covered by the font selected in the Font list. The Sample box displays a font sample that is dependent on the script selected, as well as the other font attributes. The sample preview string, which is specific to the selected character set, shows what each of the different scripts looks like.

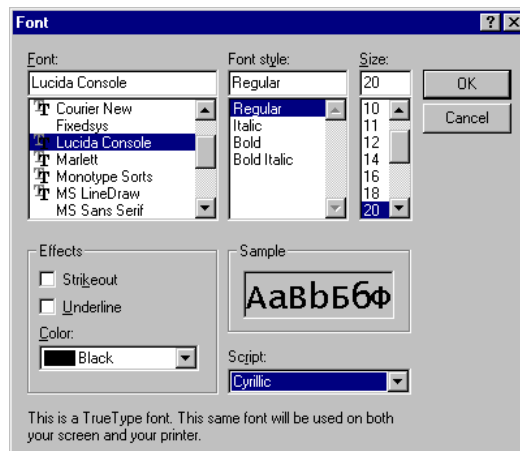


Figure 117. The Font dialog box, showing the new Font Script list

Internationally aware applications can support multilingual font selection by allowing users to select fonts via the ChooseFont common dialog box and by recognizing the

extensions to the ChooseFont data structures in Windows 95. Even Windows-based applications—which, though not originally designed for Windows 95, support formatted text but not multilingual messages—can gain some basic level of support for multilingual content. If an application uses the ChooseFont common dialog box, it benefits from the enhancements, allowing users to select from the full range of character sets and fonts configured in the system. As long as the application saves the complete logical font data structure representation for fonts, an existing Windows-based application can get by without being aware that the font selected by the user includes a possible change of character set. (Applications generally do save this data when saving text in their native format, but not all save this data when writing to interchange formats, such as RTF.)

Multilingual Support for Exchanging Information via the Clipboard

A good multilingual-aware application can exchange multilingual content with other aware applications and can exchange appropriate flat text with unaware applications, within the limitations of the ASCII text formats. Windows 95 provides special support in the data exchange APIs to pass language information along with the rich text data.

Try It!

Test Multilingual Content Support

1. In the Control Panel, open the Keyboard tool and click the Language tab. Add a couple of keyboards—for example, Swedish and French—and then click OK. At the right end of the Taskbar, notice that a small square is displayed in the status area to represent the active keyboard layout. Two letters in the square represent the language—for example, “EN” for “English.”

(If you want to write in an Eastern European language like for instance Russian, you will have to install the Multilingual language option: To install this Multilingual support, you either choose it at setup or install it later through Control Panel, Add/Remove programs, Windows setup, Accessories, Multilingual support)

2. Start a word processing application and create a document in which to test the multilingual content support.
3. Hot-switch among different input languages by pressing ALT+SHIFT, toggle through the available configured languages, and select one.
4. After you have switched to a new input language, type something. A multilingual-aware application automatically switches the font if necessary. (Of course you have to know where the keys are on that country-specific keyboard layout.)
5. Switch to a different language and type something.
6. Move the insertion point through the text. A true multilingual-aware application automatically switches the input language to match the current language format when you move through different languages in the text.

The Win32 National Language Support APIs

When users install Windows 95, they specify a locale preference. (This preference can be changed later via the Control Panel.) The Win32 NLS APIs can use either this default locale setting or a specific locale setting. Using the Win32 NLS APIs offers the following benefits to developers:

- They can easily integrate international language support into their Win32-based applications. These APIs, which are supported on both the Windows 95 and Windows NT platforms (with limited support available for Win32-based applications under Windows 3.1) allow applications to correctly retrieve regional and language settings, format date and time, sort lists according to cultural rules, compare and map strings, and determine character type information. Application developers in the U.S. can be sure that the sorting order and date formats that Microsoft provides with the operating system are correct, so all they have to do to sort or display information is use the appropriate Win32 NLS APIs.
- They can more easily develop applications for new global markets. Using this API set lowers development costs by eliminating the need for proprietary sorting methods, parsing the WIN.INI file or Registry, and locale-specific coding.

Perhaps more important for developers, the API set provides a mechanism for accurate and consistent behavior on all 32-bit Windows platforms.

Users benefit because the API set ensures that information is handled and displayed correctly for a given locale-specific format. In addition, users don't have to worry about whether their international text is being sorted properly.

