



Apple Developer Connection

Direct

# Internationalization and Localization

*(Translation: Why you should you read this article.)*

I'm not going to talk about localization or why you should have your application translated to French. Instead, I'm going to talk about something we both can relate to: money.

In 2000, almost half of Apple's revenues came from outside the U.S. So, if your application supports only U.S. English, you may be missing out on half of the market (and revenue) for your product. That's the bad news.

Here's the good news: With Mac OS X, Apple introduced new technologies to help you bring your application to international markets quickly and easily. Read on for more information and remember: If you don't get your product to foreign markets, you can always count on your competitors to do so.



## Mac OS X: An International OS

With the introduction of Mac OS X, Apple redefined what a truly international platform should be. Take a look at a clean install of Mac OS X and you'll notice support for many languages—seven

to be exact. At its introduction, Mac OS

X was released in seven languages: English, Japanese, French, German, Italian, Spanish, and Dutch. A new version of Mac OS X was released during the 2001 Worldwide Developers Conference that contains support for Korean, simplified and traditional Chinese (as well as the rest of the European languages.)

Mac OS X streamlines internationalization. From a new International pane in the System Preferences, to new technologies that help developers take advantage of Unicode™, Mac OS X streamlines internationalization. A new delivery mechanism enables devel-

opers to ship several languages in one bundle. In addition, Mac OS X includes new technologies that handle Unicode and complex scripts. Both will be discussed in more detail here. These advents will undoubtedly make Mac OS X “Le international OS.”

## Internationalization Made Easy (Translation: Why you should bundle your application.)

Mac OS X makes it easy to internationalize software and it does so in such a way that a single binary can support localizations for multiple languages and regional dialects. It also lets software developers dynamically add localized resources for new languages or regions. In Mac OS X, most software comes in the form of a bundle, of which an application package is just one type.

A bundle is an opaque directory in the file system that contains one or more executables and the resources that go with those executables. One of the primary benefits of bundles is the infrastructure they provide for localizing software. Localized resources such as image and string files, as well as Mac OS 9–style resources (rscs), can be put in bundle subdirectories whose names reflect a particular language or regional dialect (for example, Canadian French). A properly constructed Mac OS X application (or plug-in or shared library) does not hardwire paths to the resource files in these directories. Instead, when the application needs a resource, it uses a special system routine to obtain the localization that best matches the user's language preferences. (Please download “Inside Mac OS X: System Overview” if you'd like more information on Application Packaging and Bundles.). Go to:

<http://developer.apple.com/macosx/>

The new International pane in the System Preferences introduces users to a new paradigm in selection. In the following example (see fig. 1a), you'll see preferences regarding language ordering. The preferences were changed to have Japanese as the first language, French second, and English third. Next time the user logs in, the Finder will have its UI in Japanese. (Apple introduced

*Xavier Legros has been an Apple employee for four years, three of which were working as an engineer on MLTE. Most recently, he joined the ranks of Apple's Technology Managers in Worldwide Developer Relations to promote Apple technologies. He regularly drinks champagne with developers each time they adopt one of his technologies...you could be next! He can be reached at [xavier@apple.com](mailto:xavier@apple.com).*



Fig. 1a – This is the International pane from the System Preferences.

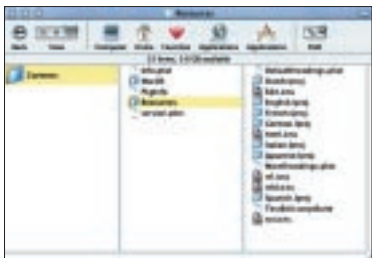


Fig. 2a – An example of what is inside a bundle. (TextEdit in this case.)

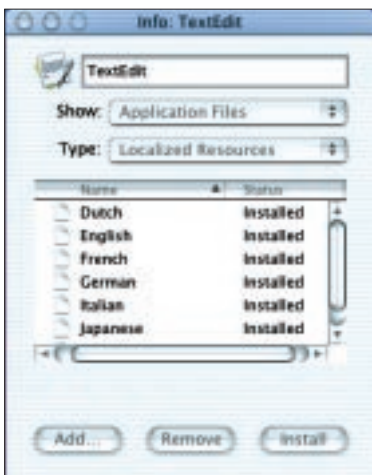


Fig. 3a – Localized languages installed as shown in the “Show Info” panel.

“Show Info” panel (CMD+I) (fig. 3a).

### Text Support in Mac OS X: International by Default Use CFString to Store Your Text

Core Foundation enables internationalization through Unicode strings and provides abstractions that contribute to operating system independence.

a new font called Hiragino with Mac OS X to display Japanese text. It looks gorgeous on the screen and in print.) The ordering of the languages means that if a user has an application that has been localized in French and English, the Finder will launch the application and load the resources in the French folder. From a developer’s point of view, if you bundle your application and have the localized resources in the appropriate folders, then you don’t have to do anything special in order to generate this behavior. The System will load the appropriate resources corresponding to the user’s preferences.

As illustrated in fig. 2a, one can see that TextEdit has been localized in seven languages. The “MacOS” folder contains the binary and “Resources” contains all the localized resources (as well as some global resources like the “icns” files.) The file “info.plist” is the heart of the bundle. It contains the version of the application, the icon to be used for the documents the application creates, as well as the application icon and many other parameters specific to the application.

You can see the localized packages supported by the application through the Finder as well when you access the

CFString and CFCharacter-Set provide a full suite of fast and efficient string manipulation and conversion functionality. String Services offer seamless Unicode support and thus greatly simplify internationalization. String Services also facilitate sharing of string data between Carbon and Cocoa applications.

### Use MLTE for Your Text Editing Needs in Carbon

MLTE, the MultiLingual Text Engine, is available on Mac OS X. MLTE is a replacement for TextEdit and is a full Unicode text engine that uses ATSUI (Apple Type Services for Unicode Imaging). When an MLTE text object is created in an application, Unicode layout is included as well as support for 2-byte scripts. No need to install the necessary TSM handlers to support Japanese, Chinese, and Korean as MLTE does this. No need to install scrollbars, drag-and-drop handlers as MLTE can provide these services as well.

MLTE is an important piece of text rendering in Mac OS X. MLTE supports the usual QuickDraw anti-aliased text but it can also render on Mac OS X with Quartz. The result gives text-editing fields the look and feel typical to Mac OS X.

You’ll find more information on MLTE at:

[http://developer.apple.com/techpubs/macosx/Carbon/text/MultilingualTextEngine/Multilingual\\_Text\\_Engine/index.html](http://developer.apple.com/techpubs/macosx/Carbon/text/MultilingualTextEngine/Multilingual_Text_Engine/index.html)

You’ll find sample code on MLTE in the CarbonLib SDK, downloadable from:

<http://connect.apple.com>

### What’s Next?

If you need resources, or help localizing an application, visit: <http://developer.apple.com/intl/> for information. You’ll find

*Continued on page ??*

## “Built for Mac OS X” Artwork Now Available

Now that customers have Mac OS X in their hands, they’ll be looking for great products to run on it. Tell the world that your product runs on Mac OS X by displaying the “Built for Mac OS X” badge on your product’s packaging. The artwork, licensing requirements, and usage guidelines are available on the ADC Software Licensing web site.

<http://developer.apple.com/mkt/swl/agreements.html#macosx>



## New Mac OS X Related Releases

Unless otherwise indicated, the following software is available from the Download Software area of the ADC Member Site at:

<http://connect.apple.com/>

- **CarbonLib 1.4a2 SDK**

The latest pre-release version of the CarbonLib 1.4 SDK for Mac OS, is now available to all ADC Members. This SDK provides all the files needed to begin Carbon development. CarbonLib 1.4 supports Mac OS 8.6 and greater.

[http://connect.apple.com](http://connect.apple.com/)

- **Mail Import Scripts 1.1**

A bundle of AppleScripts that help you import mail messages from your current email program into Mail, the Mac OS X e-mail client.

<http://asu.info.apple.com/swupdates.nsf/artnum/n12038/>

- **Mac OS X 10.0.3 Update**

This update delivers CD-burning support for iTunes; a number of improvements for overall application stability; and latest version of the Internet file transfer service (ftpd), which features important security improvements.

<http://asu.info.apple.com/swupdates.nsf/artnum/n12181/>

- **Cocoa Mailing List**

The Cocoa development mailing list is a focal point for discussions on native Mac OS X application development using the Cocoa Frameworks: Foundation and Application Kit. Cocoa is based on advanced object oriented APIs that allow development in Java and Objective-C. Subscribers to this list will be discussing frameworks, features, and technical issues specific to Cocoa application development.

<http://lists.apple.com/mailman/listinfo/cocoa-dev/>

- **Glossaries for Mac OS X**

These are translated strings of commonly used words and phrases that developers can use in their applications. Download these updated glossaries in French, German, Italian, or Spanish.

<http://developer.apple.com/intl/>

### Developer Documentation

The following new and updated documentation is available to help with successful Mac OS X application and peripheral development at:

<http://developer.apple.com/techpubs/>

Inside Mac OS X: Performance  
Carbon Documentation  
Aqua Human Interface Guidelines  
Handling Carbon Events  
Understanding Text Input and the Text Services Manager in Carbon  
Unarchiving Interface Objects With Interface Builder Services  
Interface Builder Services Reference

Text Services Manager Reference  
Multilingual Text Engine Reference  
Navigation Services Reference  
Providing User Assistance With Apple Help  
Apple Help Reference  
Carbon Event Manager  
Aqua Human Interface Guidelines  
Carbon Documentation  
Developing Cocoa Java Applications: A Tutorial  
Aqua Human Interface Guidelines  
Kernel Development  
IO Kit Fundamentals  
HID Device Interfaces  
USB Device Interfaces

### Technical Notes

TN2013 - The 'plst' Resource (Also available in Japanese)

TN2020 - Browser Plug-ins in Mac OS X

TN2002 - Compatibility between JDirect 2 and JDirect 3

### Technical Q&As

QA1036 - Displaying PCI Configuration Register contents in Open Firmware

### Sample Code

SC - Cocoa: ToolbarSample

SC - Sound: PCI Sound Input Driver

SC - Interapplication Comm: BasicInputMethod

SC - Devices and Hardware: ATA: ATADemo

## Finding Technical Information With Sherlock

Apple has created a set of Sherlock plug-ins for searching technical documentation on its Apple Developer Connection (ADC) web site. You can download these plug-ins from the web site at [http://developer.apple.com/techpubs/indexes/sherlock/Sherlock\\_Files.sit](http://developer.apple.com/techpubs/indexes/sherlock/Sherlock_Files.sit). Install the plug-ins on a Mac OS X system by moving them to the Library/Internet Search Sites/Apple directory in your home directory. Install them on a Mac OS 9 system by dragging them to the Apple folder within the Internet Search Sites folder of your System Folder.

There is one plug-in for each set of technical documentation: Carbon, QuickTime, Mac OS X, Mac OS 9, hardware, and WebObjects. These plug-ins let you perform focused searches of the latest published documentation on Apple's web site in addition to using Help Viewer to search the documentation installed on Mac OS X.

## Internationalization and Localization

*Continued from page ??*

resources for companies and organizations that can help you make the step to be international.

Remember, in order to be internationally correct in text rendering, you have to use Unicode. Unicode support in the system is achieved through MLTE and ATSUI. If you do Cocoa development you don't have to worry, all the text objects in Cocoa support Unicode layout and have built in support for Input Methods.

And last but not least, package your application! This will enhance the user experience of your customers and make your life easier.

## Did You Know?

### Guides for Internationalization

**A**pple's Technical Publications group is currently working on a book that will cover internationalization, localization, file encoding, and related issues for all Mac OS X application environments. However, you don't have to wait for that book to find out how to internationalize your application. Here are several sources of information:

- *Mac OS X: System Overview*. Includes a chapter on internationalization and multiscrypt support.
- *Project Builder Help*. Contains a section, under Files, on "Customizing for Different Regions and Platforms."
- *Cocoa Developer Documentation*. Includes a programming topic on internationalization for Cocoa applications.
- *Carbon Developer Documentation*. Includes several manuals on technologies that relate to internationalization and localization, including ATSUI, International Resources, and Text Encoding Conversion. These documents are listed under "Text and Other International Services" on the Carbon developer documentation home page.

You can access this documentation from the Help Viewer's Developer Help Center or from the ADC web site for Mac OS X documentation:

<http://developer.apple.com/techpubs/macosx/macosx.html>

## Upcoming Seminars and Events

For more information on Apple developer events please visit the developer Events page at: <http://developer.apple.com/events/>

### Training and Seminars

#### *R/com Offers Mac OS X Developer Training Online*

R/com, also known as MediaSchool <[www.mediaschool.com](http://www.mediaschool.com)>, has partnered with Apple Developer Connection to create online training for Mac OS X developers. The first courses to be released in June include "Application Development for Mac OS X," "Carbon Development for Mac OS X," and "Cocoa: The Object-Oriented Application Solution." All classes have been reviewed by Apple engineers for technical accuracy. Check out their site to take a free virtual seminar, to learn more about current and upcoming courses, and to find out about the significant discounts offered to Premier, Select, and Student members of the Apple Developer Connection.  
<http://www.mediaschool.com/adc/>

#### *Apple iServices 5-day Cocoa Training*

For application developers who want to learn how to develop Mac OS X applications using Cocoa, Apple iServices offers a five-day comprehensive, hand-on Cocoa training course. This course uses real-world examples and is perfect for developers who have a general understanding of Object-oriented concepts and practical experience with the C programming language or a C-derived language (Objective-C, Java, or C++). The course costs US \$2,495.  
<http://www.apple.com/iServices/technicaltraining/cocoadev.html>

### Developer Related Conferences

#### *1394 (FireWire) Developers' Conference 2000*

June 31-Aug 2, Redmond, WA  
Apple, Intel, and Microsoft are cosponsors  
[http://www.1394ta.org/Events/2001\\_DevCon/index.htm](http://www.1394ta.org/Events/2001_DevCon/index.htm)

#### *MACWORLD Expo, New York 2001*

July 17-20 in Jacob Javits Convention Center, NYC  
Discounted exhibitor packages available  
<http://developer.apple.com/mkt/mwny2001.html>

#### *FileMaker Developer Conference 2001*

August 12-15, Orlando, FL  
More than 40 sessions and a product showcase  
<http://www.filemaker.com/devcon/>  
Various FileMaker training classes offered concurrently  
<http://www.DevconTraining.com>