Internationalizing Data on Mac and iPhone

Deborah Goldsmith
Senior Software Engineer

Introduction

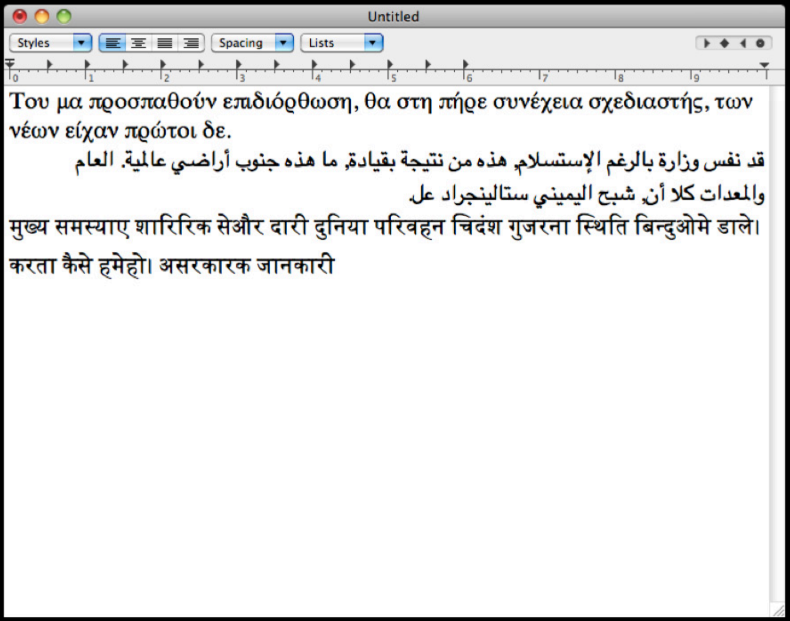
- More than half of Apple’s revenue from sales outside US
- Most of these customers don’t use English
- Many languages follow different “rules”
- Reach these customers by calling easy-to-use system APIs
- Today will be mostly concepts, not API details

Internationalization, Not Localization

- Localization: Translating your application’s UI
- Internationalization: Supporting international data
  - Text content (input, output, and processing)
  - Dates
  - Times
  - Numbers
  - Currency
  - Calendars
  - Time zones

Internationalizing Your Application

- Goal: A single application binary
  - Content in any language
  - Localizable into any language
- This session will focus on internationalization



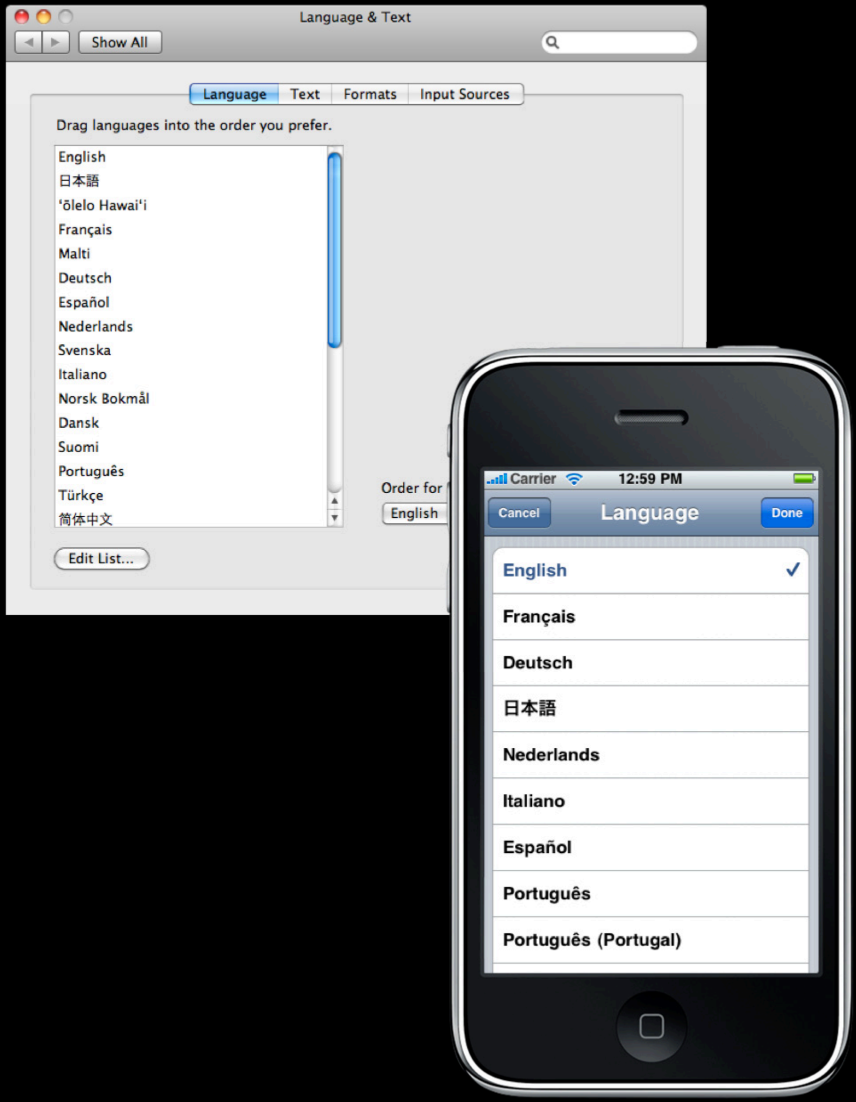
Του μα προσπαθούν επιδιόρθωση, θα στη πήρε συνέχεια σχεδιαστής, των νέων είχαν πρώτοι δε.

قد نفس وزارة بالرغم الإستسلام, هذه من نتيجة بقيادة, ما هذه جنوب أراضي عالمية. العام والمعدات كلا أن, شبح اليميني ستالينجراد عل.

मुख्य समस्याए शारिरिक सेऔर दारी दुनिया परिवहन चिदंश गुजरना स्थिति बिन्दुओमे डाले। करता कैसे हमेहो। असरकारक जानकारी

Content

- Produced by the user
- Can be in multiple languages
- Can be a mix of supported languages
- Not our focus today


Language & Text

Show All

Language | Text | Formats | Input Sources

Drag languages into the order you prefer.

English
日本語
ʻōlelo Hawaiʻi
Français
Malti
Deutsch
Español
Nederlands
Svenska
Italiano
Norsk Bokmål
Dansk
Suomi
Português
Türkçe
简体中文

Order for
English

Edit List...

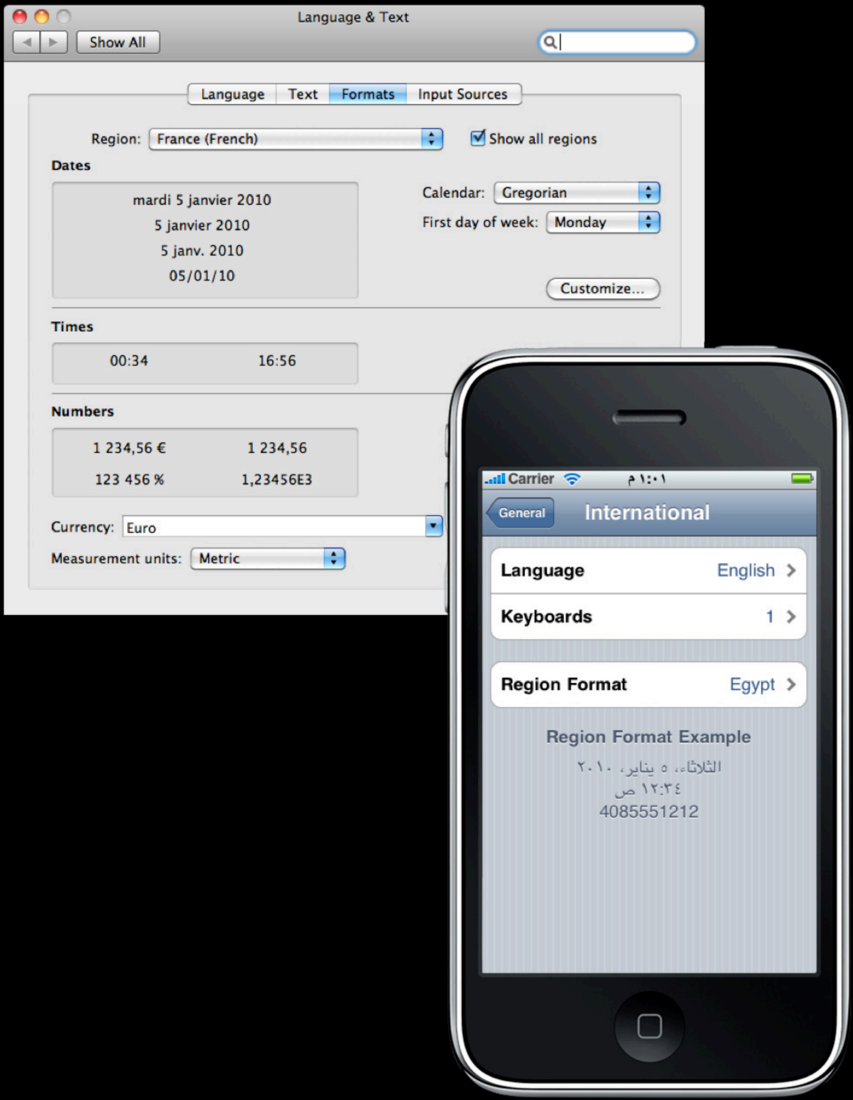
Carrier 12:59 PM

Cancel Language Done

English ✓
Français
Deutsch
日本語
Nederlands
Italiano
Español
Português
Português (Portugal)

Language

- Controls language of UI (localization)
  - Selects one *lang*.lproj in app or framework bundle
- Controls sort order and word breaking
- Change requires restarting applications
- Not our focus today



Region: France (French)

☑ Show all regions

Dates

mardi 5 janvier 2010
5 janvier 2010
5 janv. 2010
05/01/10

Calendar: Gregorian

First day of week: Monday

Times

00:34 16:56

Numbers

1 234,56 € 1 234,56
123 456 % 1,23456E3

Currency: Euro

Measurement units: Metric

١:٠١ م

Language English

Keyboards 1

Region Format Egypt

Region Format Example

الثلاثاء، ٥ يناير، ٢٠١٠
١٢:٣٤ ص
4085551212

Locale/Region

- Controls dates, times, numbers
- Has language component, which *usually* matches UI language
- Can change without relaunching applications
- Our focus for today

Dramatis Personæ

NSLocale	Current region, formats, and other properties
NSNumberFormatter	Formats and parses numbers
NSDateFormatter	Formats and parses dates and times
NSCalendar	Current calendar and associated operations
NSTimeZone	Current time zone and associated operations
NSString	Sorting, searching, and more

NSLocale

- Set by the “Region Format” preference
- Example: US English identifier is en_US
- Complex example: sr_Latn_RS@currency=EUR
- Identifier consists of:
  - Language (always)
  - Region (almost always)
  - Script (sometimes)
  - Variant (occasionally)
  - Keywords (for overrides)

NSLocale

- Set by the "Region Format" preference
- Example: US English identifier is en_US
- Complex example: sr_Latn_RS@currency=EUR
- Identifier consists of:
    - Language (always)
    - Region (almost always)
    - Script (sometimes)
    - Variant (occasionally)
    - Keywords (for overrides)

NSLocale

- Set by the “Region Format” preference
- Example: US English identifier is en_US
- Complex example: sr_Latn_RS@currency=EUR
- Identifier consists of:
  - Language (always)
  - Region (almost always)
  - Script (sometimes)
  - Variant (occasionally)
  - Keywords (for overrides)

NSLocale

- Set by the “Region Format” preference
- Example: US English identifier is en_US
- Complex example: sr_Latn_RS@currency=EUR
- Identifier consists of:
  - Language (always)
  - Region (almost always)
  - Script (sometimes)
  - Variant (occasionally)
  - Keywords (for overrides)

NSLocale

- Set by the “Region Format” preference
- Example: US English identifier is en_US
- Complex example: sr_Latn_RS@currency=EUR
- Identifier consists of:
  - Language (always)
  - Region (almost always)
  - Script (sometimes)
  - Variant (occasionally)
  - Keywords (for overrides)

Getting a Locale

- [NSLocale currentLocale]
    - Doesn’t change after creation
- NSCurrentLocaleDidChangeNotification
- [NSLocale autoupdatingCurrentLocale]
    - Tracks preferences, but possible race condition
    - Affects objects set to that locale

Numbers

Differences between locales

Decimal point and grouping separator, size	1,234.56	1 234,56
Digits (not all use 0–9)	1,234.56	١٬٢٣٤٫٥٦
Currency	$123.45	123.45 €
Percentage	45%	٤٥٪
NaN, ∞, etc.	NaN	EiTa

Number Formatting (Simple)

[NSNumberFormatter localizedStringFromNumber:numberStyle:]

General	1,234.56
Currency	$1,234.56
Percentage	123,456%
Scientific	1.23456E+03

Number Formatting (Advanced)

- Create an NSNumberFormatter and keep it around if:
- Formatting many numbers
- Parsing numbers
- Need fine control of format
  - Digits
  - Fraction
  - Sign
  - etc.

Common Errors

- Formatting numbers with stringWithFormat:, printf, scanf
  - %d, %f will not handle non-ASCII digits correctly
- Assuming decimal point or grouping separator
- Assuming percent format or character
- Erasing locale information by setting the pattern
- Forgetting about currency conversion

Dates and Times

Differences between locales

Language of months, days, AM/PM, today	jeudi 10 juin 2010
Calendar in use	平成22年6月10日
First day of week	Monday vs. Sunday
12 vs. 24 hour time	2:15 PM vs. 14:15
Format, order of dates and times	10 June 2010

Date and Time Formatting

NSDateFormatter predefined formats

Short	6/10/10	4:00 PM
Medium	Jun 10, 2010	4:00:04 PM
Long	June 10, 2010	4:00:04 PM PDT
Full	Thursday, June 10, 2010	4:00:04 PM Pacific Daylight Time

New Options

New

- [NSDateFormatter setDoesRelativeDateFormatting:]
  - “June 5, 2010 4:00 PM” → “Yesterday 4:00 PM”
- [NSDateFormatter dateFormatFromTemplate:options:locale:]
  - Returns format string appropriate for locale; use with instance

Template string	Sample format strings	Sample results
j	h a H	4 PM 16
yMMMM	MMMM y Gy年M月	June 2010 平成22年6月

Common Errors

- Using NSDateFormatter for parsing/formatting Internet dates without setting locale and considering time zone
  - Internet dates are not localized
  - `[formatter setLocale:[[[NSLocale alloc] initWithIdentifier:@”en_US_POSIX”] autorelease]];`
  - strptime_l and strftime_l (pass NULL for locale)
- Parsing format string for “separators”
  - dateFormatFromTemplate: Solves most such problems
- Using NSCalendarDate at all, or using NSDate for parsing/formatting
- Assuming Gregorian calendar

Calendars

Differences between calendars

Year	2553
Era	Heisei 22
Number of months per year	12, 13, variable
Lengths of months	From 5 to 31 days
When years change	昭和64年1月7日 → 平成1年1月8日

Calendar Operations

- NSCalendar abstracts calendar operations, date computations
  - Days in month
  - Months in year
  - Calendar components for absolute time
  - Add three days
- Mac OS X 10.6 supports non-Gregorian calendars
- iPhone OS 4.0 supports “Gregorian-like” non-Gregorian calendars

Common Errors

- Assuming Gregorian calendar
- Assuming 12 months per year
- Assuming month numbers are sequential
- Assuming day numbers are sequential
- Assuming era is optional
- Assuming weeks start on Sunday
- Assuming years can only change on first day of first month of year
- Ignoring calendar when defining recurrences

Time Zones

- Differences between time zones
  - Offset from GMT/UTC
  - Rules for daylight time
  - Unique identifier: Olson ID
  - User-visible localized names
- NSTimeZone abstracts time zone operations
  - Current offset from GMT
  - Offset from GMT at particular absolute time
  - Time of offset transition
  - Localized names

Common Errors

- Assuming GMT offset, rules in use
  - Arizona, Indiana, Mexico…
- Showing Olson ID to user (America/Los_Angeles)
  - Use localized name (Pacific Time)
- Displaying wrong time zone name
  (Pacific Time vs. Pacific Daylight Time vs. Pacific Standard Time)
- Assuming abbreviations (e.g. “PST”) are unique

Natural Language and NSString

- Breaking
  - Use [NSString enumerateSubstringsInRange:options:usingBlock:] to perform lexical operations
  - Word boundaries, line boundaries, sentence boundaries, …
- Sorting
  - Different languages have different sort orders
  - Diacritic (accent) significance and handling vary between languages
  - Use [NSString localizedStandardCompare:]

Sorting Examples

Hawaiian

telanuali
lāpule
malaki
pepeluali
poʻakahi
poʻalua
bad
Bad
black-bird
black-birds
blackbird
blackbirds
cote

French

boef
Boef
bœf
deja
dejà
déjà
mardi
Meme
même
Mémé
pêche
pèché
pêché

Common Errors

- Assuming words and lines are separated by whitespace
  - Not true for Japanese, Chinese, Thai, etc.
- Using [NSString compare:] for user-visible sorts
  - This comparison is *not localized*; use [NSString localizedStandardCompare:]
  - [NSString compare:options:range:locale:] OK for advanced use
- Using diacritic- or case-insensitivity for sorting
  - Diacritic- and case-insensitivity are for *searching*, not *sorting*
  - Some languages sort uppercase first, others lowercase

Related Sessions

Advanced Text Handling for iPhone OS	Nob Hill Tuesday 4:30PM
Understanding Foundation	Russian Hill Thursday 9:00AM

Labs

Internationalization Lab	Application Frameworks Lab C Thursday 11:30AM–1:45PM

More Information

Apple Developer Forums
http://devforums.apple.com

Q&A

 WWDC10