CFLocale Reference

Core Foundation

2007-05-23

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Simultaneously published in the United States and Canada.

Contents

CFLocale Reference

Derived From:	CFType
Framework:	CoreFoundation/CoreFoundation.h
Declared in	CFLocale.h
Companion guides	Locales Programming Guide Internationalization Programming Topics

Overview

Unicode operations such as collation and text boundary determination can be affected by the conventions of a particular language or region. CFLocale objects specify language-specific or region-specific information for locale-sensitive operations.

The CFLocale opaque type provides support for obtaining available locales, obtaining localized locale names, and converting among locale data formats. Locale identifiers in Mac OS X follow the IETF's BCP 47. CFLocale never uses Script Manager codes (except for the legacy support provided by `CFLocaleCreateCanonicalLocaleIdentifierFromScriptManagerCodes` (page 11))—the Script Manager and all its concepts are deprecated.

For more information on locale identifiers and the use of CFLocale, see *Locales Programming Guide*. It is also useful to read the ICU's User Guide for the Locale Class.

CFLocale is "toll-free bridged" with its Cocoa Foundation counterpart, NSLocale. This means that the Core Foundation type is interchangeable in function or method calls with the bridged Foundation object. Therefore, in a method where you see an `NSLocale *` parameter, you can pass in a `CFLocaleRef`, and in a function where you see a `CFLocaleRef` parameter, you can pass in an `NSLocale` instance. See Interchangeable Data Types for more information on toll-free bridging.

Functions by Task

Creating a Locale

`CFLocaleCopyCurrent` (page 8)
Returns a copy of the logical locale for the current user.

`CFLocaleCreate` (page 10)
Creates a locale for the given arbitrary locale identifier.

`CFLocaleCreateCopy` (page 13)
Returns a copy of a locale.

`CFLocaleGetSystem` (page 15)
Returns the root, canonical locale.

Getting System Locale Information

`CFLocaleCopyAvailableLocaleIdentifiers` (page 7)
Returns an array of CFString objects that represents all locales for which locale data is available.

Getting ISO Information

`CFLocaleCopyISOCountryCodes` (page 9)
Returns an array of CFString objects that represents all known legal ISO country codes.

`CFLocaleCopyISOLanguageCodes` (page 10)
Returns an array of CFString objects that represents all known legal ISO language codes.

`CFLocaleCopyISOCurrencyCodes` (page 9)
Returns an array of CFString objects that represents all known legal ISO currency codes.

`CFLocaleCopyCommonISOCurrencyCodes` (page 7)
Returns an array of strings that represents ISO currency codes for currencies in common use.

Language Preferences

`CFLocaleCopyPreferredLanguages` (page 10)
Returns the array of canonicalized locale IDs that the user prefers.

Getting Information About a Locale

`CFLocaleCopyDisplayNameForPropertyValue` (page 8)
Returns the display name for the given value.

`CFLocaleGetValue` (page 15)
Returns the corresponding value for the given key of a locale's key-value pair.

`CFLocaleGetIdentifier` (page 14)
Returns the given locale's identifier.

Getting and Creating Locale Identifiers

`CFLocaleCreateCanonicalLocaleIdentifierFromScriptManagerCodes` (page 11)
Returns a canonical locale identifier from given language and region codes.

`CFLocaleCreateCanonicalLanguageIdentifierFromString` (page 11)
Returns a canonical language identifier by mapping an arbitrary locale identification string to the canonical identifier

`CFLocaleCreateCanonicalLocaleIdentifierFromString` (page 12)
Returns a canonical locale identifier by mapping an arbitrary locale identification string to the canonical identifier.

`CFLocaleCreateComponentsFromLocaleIdentifier` (page 13)
Returns a dictionary containing the result from parsing a locale ID consisting of language, script, country, variant, and keyword/value pairs.

`CFLocaleCreateLocaleIdentifierFromComponents` (page 14)
Returns a locale identifier consisting of language, script, country, variant, and keyword/value pairs derived from a dictionary containing the source information.

Getting the CFLocale Type ID

`CFLocaleGetTypeID` (page 15)
Returns the type identifier for the CFLocale opaque type.

Functions

CFLocaleCopyAvailableLocaleIdentifiers

Returns an array of CFString objects that represents all locales for which locale data is available.

```
CFArrayRef CFLocaleCopyAvailableLocaleIdentifiers (
   void
);
```

Return Value
An array of CFString objects that represents all locales for which locale data is available. Ownership follows the Create Rule.

Availability
Available in Mac OS X v10.4 and later.

Declared In
`CFLocale.h`

CFLocaleCopyCommonISOCurrencyCodes

Returns an array of strings that represents ISO currency codes for currencies in common use.

```
CFArrayRef CFLocaleCopyCommonISOCurrencyCodes (
   void
);
```

Return Value
An array of CFString objects that represents ISO currency codes for currencies in common use. Ownership follows the Create Rule.

Availability
Available in Mac OS X v10.5 and later.

Declared In
`CFLocale.h`

CFLocaleCopyCurrent

Returns a copy of the logical locale for the current user.

```
CFLocaleRef CFLocaleCopyCurrent (
   void
);
```

Return Value
The logical locale for the current user that is formed from the settings for the current user's chosen system locale overlaid with any custom settings the user has specified in System Preferences. May return a retained cached object, not a new object. Ownership follows the Create Rule.

Discussion
Settings you get from this locale do not change as a user's preferences are changed so that your operations are consistent. Typically you perform some operations on the returned object and then release it. Since the returned object may be cached, you do not need to hold on to it indefinitely.

Note that locale settings are independent of the user's language setting. The language of the current locale may not correspond to the language at the first index in the `AppleLanguages` array from user defaults. For more details, see Locale Concepts in *Locales Programming Guide*; see also `CFLocaleCopyPreferredLanguages` (page 10).

Availability
Available in Mac OS X v10.3 and later.

Related Sample Code
CFFTPSample
CFPrefTopScores

Declared In
`CFLocale.h`

CFLocaleCopyDisplayNameForPropertyValue

Returns the display name for the given value.

```
CFStringRef CFLocaleCopyDisplayNameForPropertyValue (
   CFLocaleRef displayLocale,
   CFStringRef key,
   CFStringRef value
);
```

Parameters

displayLocale
A locale object.

key
A string that identifies the type that *value* is. It must be one of the standard locale property keys (see "Locale Property Keys" (page 16)).

value
The value for which the display name is required.

Return Value
The display name for *value*. Returns NULL if there was a problem creating the object. Ownership follows the Create Rule.

Discussion
Note that not all locale property keys have values with display name values.

Availability
Available in Mac OS X v10.4 and later.

Declared In
CFLocale.h

CFLocaleCopyISOCountryCodes

Returns an array of CFString objects that represents all known legal ISO country codes.

```
CFArrayRef CFLocaleCopyISOCountryCodes (
   void
);
```

Return Value
An array of CFString objects that represents all known legal ISO country codes. Ownership follows the Create Rule.

Discussion
Note: many of these will not have any supporting locale data in Mac OS X.

Availability
Available in Mac OS X v10.4 and later.

Declared In
CFLocale.h

CFLocaleCopyISOCurrencyCodes

Returns an array of CFString objects that represents all known legal ISO currency codes.

```
CFArrayRef CFLocaleCopyISOCurrencyCodes (
   void
);
```

Return Value
An array of CFString objects that represents all known legal ISO currency codes.Ownership follows the Create Rule.

Discussion
Note: many of these will not have any supporting locale data in Mac OS X.

Availability
Available in Mac OS X v10.4 and later.

Declared In
CFLocale.h

CFLocaleCopyISOLanguageCodes

Returns an array of CFString objects that represents all known legal ISO language codes.

```
CFArrayRef CFLocaleCopyISOLanguageCodes (
   void
);
```

Return Value
An array of CFString objects that represents all known legal ISO language codes. Ownership follows the Create Rule.

Discussion
Note: many of these will not have any supporting locale data in Mac OS X.

Availability
Available in Mac OS X v10.4 and later.

Declared In
CFLocale.h

CFLocaleCopyPreferredLanguages

Returns the array of canonicalized locale IDs that the user prefers.

```
CFArrayRef CFLocaleCopyPreferredLanguages (
    void
);
```

Return Value
The array of canonicalized CFString locale IDs that the current user prefers. Ownership follows the Create Rule.

Availability
Available in Mac OS X v10.5 and later.

Declared In
CFLocale.h

CFLocaleCreate

Creates a locale for the given arbitrary locale identifier.

```
CFLocaleRef CFLocaleCreate (
   CFAllocatorRef allocator,
   CFStringRef localeIdentifier
);
```

Parameters

allocator

The allocator to use to allocate memory for the new object. Pass `NULL` or `kCFAllocatorDefault` to use the current default allocator.

localeIdentifier

A string representation of an arbitrary locale identifier.

Return Value

A new locale that corresponds to the arbitrary locale identifier *localeIdentifier*. Returns `NULL` if there was a problem creating the object. Ownership follows the Create Rule.

Availability

Available in Mac OS X v10.3 and later.

Declared In

`CFLocale.h`

CFLocaleCreateCanonicalLanguageIdentifierFromString

Returns a canonical language identifier by mapping an arbitrary locale identification string to the canonical identifier

```
CFStringRef CFLocaleCreateCanonicalLanguageIdentifierFromString (
   CFAllocatorRef allocator,
   CFStringRef localeIdentifier
);
```

Parameters

allocator

The allocator to use to allocate memory for the new object. Pass `NULL` or `kCFAllocatorDefault` to use the current default allocator.

localeIdentifier

A string representation of an arbitrary locale identifier.

Return Value

A string that represents the canonical language identifier for the specified arbitrary locale identifier. Returns `NULL` if there was a problem creating the object. Ownership follows the Create Rule.

Availability

Available in Mac OS X v10.4 and later.

Declared In

`CFLocale.h`

CFLocaleCreateCanonicalLocaleIdentifierFromScriptManagerCodes

Returns a canonical locale identifier from given language and region codes.

```
CFStringRef CFLocaleCreateCanonicalLocaleIdentifierFromScriptManagerCodes (
   CFAllocatorRef allocator,
   LangCode lcode,
   RegionCode rcode
);
```

Parameters

allocator

The allocator to use to allocate memory for the new object. Pass `NULL` or `kCFAllocatorDefault` to use the current default allocator.

lcode

A Mac OS X language code.

rcode

A Mac OS X region code.

Return Value

A canonical locale identifier created by mapping *lcode* and *rcode* to a locale. Returns `NULL` if there was a problem creating the object. Ownership follows the Create Rule.

Availability

Available in Mac OS X v10.3 and later.

Declared In

`CFLocale.h`

CFLocaleCreateCanonicalLocaleIdentifierFromString

Returns a canonical locale identifier by mapping an arbitrary locale identification string to the canonical identifier.

```
CFStringRef CFLocaleCreateCanonicalLocaleIdentifierFromString (
   CFAllocatorRef allocator,
   CFStringRef localeIdentifier
);
```

Parameters

allocator

The allocator to use to allocate memory for the new object. Pass `NULL` or `kCFAllocatorDefault` to use the current default allocator.

localeIdentifier

A string representation of an arbitrary locale identifier (for example, “English”).

Return Value

A canonical locale identifier created by mapping the arbitrary locale identification string to the canonical identifier for the corresponding locale (for example, “en”). Returns `NULL` if there was a problem creating the object. Ownership follows the Create Rule.

Availability

Available in Mac OS X v10.3 and later.

Declared In

`CFLocale.h`

CFLocaleCreateComponentsFromLocaleIdentifier

Returns a dictionary containing the result from parsing a locale ID consisting of language, script, country, variant, and keyword/value pairs.

```
CFDictionaryRef CFLocaleCreateComponentsFromLocaleIdentifier (
   CFAllocatorRef allocator,
   CFStringRef localeID
);
```

Parameters

allocator

The allocator to use to allocate memory for the new object. Pass `NULL` or `kCFAllocatorDefault` to use the current default allocator.

localeID

The locale ID to be used when creating the locale dictionary.

Return Value

A dictionary containing the result from parsing a locale ID consisting of language, script, country, variant, and keyword/value pairs. Returns `NULL` if there was a problem creating the object. Ownership follows the Create Rule.

Discussion

The dictionary keys are the constant CFString objects that correspond to the locale ID components; the values correspond to constants where available. For example: the string "en_US@calendar=japanese" yields a dictionary with three entries: `kCFLocaleLanguageCode=en`, `kCFLocaleCountryCode=US`, and `kCFLocaleCalendarIdentifier=kCFJapaneseCalendar`. See also `CFLocaleCreateLocaleIdentifierFromComponents` (page 14).

Availability

Available in Mac OS X v10.4 and later.

Declared In

`CFLocale.h`

CFLocaleCreateCopy

Returns a copy of a locale.

```
CFLocaleRef CFLocaleCreateCopy (
   CFAllocatorRef allocator,
   CFLocaleRef locale
);
```

Parameters

allocator

The allocator to use to allocate memory for the new object. Pass `NULL` or `kCFAllocatorDefault` to use the current default allocator.

locale

The locale object to copy.

Return Value

A new locale that is a copy of *locale*. Returns `NULL` if there was a problem creating the object. Ownership follows the Create Rule.

Availability
Available in Mac OS X v10.3 and later.

Declared In
`CFLocale.h`

CFLocaleCreateLocaleIdentifierFromComponents

Returns a locale identifier consisting of language, script, country, variant, and keyword/value pairs derived from a dictionary containing the source information.

```
CFStringRef CFLocaleCreateLocaleIdentifierFromComponents (
   CFAllocatorRef allocator,
   CFDictionaryRef dictionary
);
```

Parameters

allocator

The allocator to use to allocate memory for the new object. Pass `NULL` or `kCFAllocatorDefault` to use the current default allocator.

dictionary

The dictionary to use when creating the locale identifier.

Return Value
A locale identifier consisting of language, script, country, variant, and keyword/value pairs derived from *dictionary*. Returns `NULL` if there was a problem creating the string. Ownership follows the Create Rule.

Discussion
Reverses the actions of `CFLocaleCreateComponentsFromLocaleIdentifier` (page 13), creating a single string from the data in the specified dictionary. For example, the dictionary {`kCFLocaleLanguageCode=en, kCFLocaleCountryCode=US, kCFLocaleCalendarIdentifier=kCFJapaneseCalendar}` becomes `"en_US@calendar=japanese"`.

Availability
Available in Mac OS X v10.4 and later.

Declared In
`CFLocale.h`

CFLocaleGetIdentifier

Returns the given locale's identifier.

```
CFStringRef CFLocaleGetIdentifier (
   CFLocaleRef locale
);
```

Parameters

locale

The locale object to examine.

Return Value
A string representation of *locale*'s identifier. This may not be the same string that was used to create the locale—it may be canonicalized. Ownership follows the Get Rule.

Availability
Available in Mac OS X v10.3 and later.

Declared In
`CFLocale.h`

CFLocaleGetSystem

Returns the root, canonical locale.

```
CFLocaleRef CFLocaleGetSystem (
   void
);
```

Return Value
The root, canonical locale. Ownership follows the Get Rule.

Discussion
The root locale contains fixed backstop settings for all locale information.

Availability
Available in Mac OS X v10.3 and later.

Related Sample Code
LSMSmartCategorizer

Declared In
`CFLocale.h`

CFLocaleGetTypeID

Returns the type identifier for the CFLocale opaque type.

```
CFTypeID CFLocaleGetTypeID (
   void
);
```

Return Value
The type identifier for the CFLocale opaque type.

Availability
Available in Mac OS X v10.3 and later.

Declared In
`CFLocale.h`

CFLocaleGetValue

Returns the corresponding value for the given key of a locale's key-value pair.

```
CFTypeRef CFLocaleGetValue (
   CFLocaleRef locale,
   CFStringRef key
);
```

Parameters

locale

The locale object to examine.

key

The key for which to obtain the corresponding value. Possible values are described in "Locale Property Keys" (page 16).

Return Value

The value corresponding to the given key in locale. The value may be any type of CFType object. Ownership follows the Get Rule.

Discussion

Locale objects use key-value pairs to store property values. Use this function to get the value of a specific property.

Availability

Available in Mac OS X v10.3 and later.

Declared In

`CFLocale.h`

Data Types

CFLocaleRef

A reference to a CFLocale object.

```
typedef const struct __CFLocale *CFLocaleRef;
```

Availability

Available in Mac OS X version 10.3 and later.

Declared In

`CFLocale.h`

Constants

Locale Property Keys

Predefined locale keys used to get property values.

```
const CFStringRef kCFLocaleMeasurementSystem;
const CFStringRef kCFLocaleDecimalSeparator;
const CFStringRef kCFLocaleGroupingSeparator;
const CFStringRef kCFLocaleCurrencySymbol;
const CFStringRef kCFLocaleCurrencyCode;

const CFStringRef kCFLocaleIdentifier;
const CFStringRef kCFLocaleLanguageCode;
const CFStringRef kCFLocaleCountryCode;
const CFStringRef kCFLocaleScriptCode;
const CFStringRef kCFLocaleVariantCode;
const CFStringRef kCFLocaleExemplarCharacterSet;
const CFStringRef kCFLocaleCalendarIdentifier;
const CFStringRef kCFLocaleCalendar;
const CFStringRef kCFLocaleCollationIdentifier;
const CFStringRef kCFLocaleUsesMetricSystem;
```

Constants

`kCFLocaleMeasurementSystem`

Specifies the measurement system used.

The corresponding value is a CFString, for example “Metric” or “U.S.”.

Available in Mac OS X v10.3 and later.

Declared in `CFLocale.h`.

`kCFLocaleDecimalSeparator`

Specifies the decimal point string.

The corresponding value is a CFString, for example “.” or “,”.

Available in Mac OS X v10.3 and later.

Declared in `CFLocale.h`.

`kCFLocaleGroupingSeparator`

Specifies the separator string between groups of digits.

The corresponding value is a CFString, for example “,” or “.”.

Available in Mac OS X v10.3 and later.

Declared in `CFLocale.h`.

`kCFLocaleCurrencySymbol`

Specifies the currency symbol.

The corresponding value is a CFString, for example “$” or “£”.

Available in Mac OS X v10.3 and later.

Declared in `CFLocale.h`.

`kCFLocaleCurrencyCode`

Specifies the locale currency code.

The corresponding value is a CFString, for example “USD” or “GBP”.

Available in Mac OS X v10.3 and later.

Declared in `CFLocale.h`.

`kCFLocaleIdentifier`
Specifies locale identifier.

The corresponding value is a CFString containing the POSIX locale identifier as used by ICU, such as "`ja_JP`". If you have a variant locale or a different currency or calendar, it can be as complex as "`en_US_POSIX@calendar=japanese;currency=EUR`" or "`az_Cyrl_AZ@calendar=buddhist;currency=JPY`".

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFLocaleLanguageCode`
Specifies the locale language code.

The corresponding value is a CFString containing an ISO 639-x/IETF BCP 47 language identifier, such as "`ja`".

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFLocaleCountryCode`
Specifies the locale country code.

The corresponding value is a CFString containing an ISO county code, such as "`JP`".

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFLocaleScriptCode`
Specifies the locale script code.

The corresponding value is a CFString containing a Unicode script tag (strictly, an ISO 15924 script tag). Usually this is empty (it is for "`ja_JP`"). It may be present for locales where a script *must* be specified, for example "`uz-Latn-UZ`" vs. "`uz-Cyrl-UZ`" for Uzbek in Latin vs. Cyrillic (in the first case the script code is "`Latn`", and in the second it is "`Cyrl`").

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFLocaleVariantCode`
Specifies the locale variant code.

The corresponding value is a CFString containing the variant name. The variant code is arbitrary and application-specific. ICU adds "`_EURO`" to its locale designations for locales that support the Euro currency. For "`en_US_POSIX`" the variant is "`POSIX`", and for "`hy_AM_REVISED`" it is "`REVISED`".

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFLocaleExemplarCharacterSet`
Specifies the locale character set.

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFLocaleCalendarIdentifier`
Specifies the locale calendar identifier.

The corresponding value is a CFString containing the calendar identifier (for possible values, see "Locale Calendar Identifiers" (page 19)).

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFLocaleCalendar`
Specifies the locale calendar.

The corresponding value is a CFCalendar.

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFLocaleCollationIdentifier`
Specifies the locale collation identifier.

The corresponding value is a collation.

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFLocaleUsesMetricSystem`
Specifies the whether the locale uses the metric system.

The corresponding value is a CFBoolean.

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

Discussion
Locale objects use key-value pairs to store property values. Use the `CFLocaleGetValue` (page 15) function to get the value of a specific property listed above.

Locale Calendar Identifiers

Predefined locale keys used to get calendar values—values for `kCFLocaleCalendarIdentifier`.

```
const CFStringRef kCFGregorianCalendar;
const CFStringRef kCFBuddhistCalendar;
const CFStringRef kCFChineseCalendar;
const CFStringRef kCFHebrewCalendar;
const CFStringRef kCFIslamicCalendar;
const CFStringRef kCFIslamicCivilCalendar;
const CFStringRef kCFJapaneseCalendar;
```

Constants
`kCFGregorianCalendar`
Specifies the Gregorian calendar.

Available in Mac OS X v10.3 and later.

Declared in `CFLocale.h`.

`kCFBuddhistCalendar`
Specifies the Buddhist calendar.

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFChineseCalendar`
Specifies the Chinese calendar.

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFHebrewCalendar`
Specifies the Hebrew calendar.

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFIslamicCalendar`
Specifies the Islamic calendar.

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFIslamicCivilCalendar`
Specifies the Islamic Civil calendar.

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

`kCFJapaneseCalendar`
Specifies the Japanese calendar.

Available in Mac OS X v10.4 and later.

Declared in `CFLocale.h`.

Discussion
Locale objects use key-value pairs to store property values. Use the `CFLocaleGetValue` (page 15) function to get the value of a specific property listed above.

Locale Change Notification

Identifier for notification sent if the current locale changes.

```
const CFStringRef kCFLocaleCurrentLocaleDidChangeNotification
```

Constants
`kCFLocaleCurrentLocaleDidChangeNotification`
Identifier for the notification sent if the current locale changes.

This is a local notification posted when the user changes locale information in the System Preferences panel. Keep in mind that there is no order in how notifications are delivered to observers; frameworks or other parts of your code may also be observing this notification to take their own actions, and these may not have occurred at the time you receive the notification.

There is no object or user info for this notification.

Available in Mac OS X v10.5 and later.

Declared in `CFLocale.h`.

Declared In
`CFLocale.h`

Document Revision History

This table describes the changes to *CFLocale Reference*.

Date	Notes
2007-05-23	Corrected minor typographical errors.
2007-03-06	Clarified values for locale properties.
2006-12-05	Clarified the relationship of CFLocaleCopyCurrent to the user defaults AppleLanguages array.
2006-03-08	Added the definition of the CFLocaleCreateCanonicalLocaleIdentifier-FromScriptManagerCodes function.
2006-01-10	Corrected a typographical error.
2005-12-06	Made corrections in Companion Documents list.
2005-11-09	Added kCFGregorianCalendar constant. Added availability information to constant definitions.
2005-08-11	Corrected function declaration for CFLocaleCopyDisplayNameForPropertyValue.
2005-04-29	Updated to include new API for Mac OS X version 10.4.
2004-05-27	Fixed minor typographical error in and added example to `CFLocaleCreateCanonicalLocaleIdentifierFromString` (page 12); augmented description of `CFLocaleCopyCurrent` (page 8).
2003-10-24	Included links to conceptual reference.
2003-07-01	First version of this document.

Document Revision History

Index

C

K

L