
CTFontDescriptor Reference

[Carbon](#) > [Text & Fonts](#)



2008-10-15



Apple Inc.
© 2008 Apple Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

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

Apple, the Apple logo, Carbon, Mac, Mac OS, and Macintosh are trademarks of Apple Inc., registered in the United States and other countries.

Helvetica is a registered trademark of Heidelberger Druckmaschinen AG, available from Linotype Library GmbH.

Simultaneously published in the United States and Canada.

Even though Apple has reviewed this document, **APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE**

ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

CTFontDescriptor Reference 5

Overview	5
Functions by Task	5
Creating Font Descriptors	5
Getting Attributes	6
Getting the Font Descriptor Type	6
Functions	6
CTFontDescriptorCopyAttribute	6
CTFontDescriptorCopyAttributes	6
CTFontDescriptorCopyLocalizedAttribute	7
CTFontDescriptorCreateCopyWithAttributes	8
CTFontDescriptorCreateCopyWithFeature	8
CTFontDescriptorCreateCopyWithVariation	9
CTFontDescriptorCreateMatchingFontDescriptor	9
CTFontDescriptorCreateMatchingFontDescriptors	10
CTFontDescriptorCreateWithAttributes	10
CTFontDescriptorCreateWithNameAndSize	11
CTFontDescriptorGetTypeID	11
Data Types	12
CTFontDescriptorRef	12
Constants	12
Font Attributes	12
Font Traits	16

Document Revision History 21

Index 23

CTFontDescriptor Reference

Derived From:	CType
Framework:	ApplicationServices/CoreText
Declared in	CTFontDescriptor.h CTFontTraits.h

Overview

The `CTFontDescriptor` opaque type represents a font descriptor, that is, a dictionary of attributes (such as name, point size, and variation) that can completely specify a font.

A font descriptor can be an incomplete specification, in which case the system chooses the most appropriate font to match the given attributes.

Functions by Task

Creating Font Descriptors

[CTFontDescriptorCreateWithNameAndSize](#) (page 11)

Creates a new font descriptor with the provided PostScript name and size.

[CTFontDescriptorCreateWithAttributes](#) (page 10)

Creates a new font descriptor reference from a dictionary of attributes.

[CTFontDescriptorCreateCopyWithAttributes](#) (page 8)

Creates a copy of the original font descriptor with new attributes.

[CTFontDescriptorCreateCopyWithVariation](#) (page 9)

Creates a copy of the original font descriptor with a new variation instance.

[CTFontDescriptorCreateCopyWithFeature](#) (page 8)

Copies a font descriptor with new feature settings.

[CTFontDescriptorCreateMatchingFontDescriptors](#) (page 10)

Returns an array of normalized font descriptors matching the provided descriptor.

[CTFontDescriptorCreateMatchingFontDescriptor](#) (page 9)

Returns the single preferred matching font descriptor based on the original descriptor and system precedence.

Getting Attributes

[CTFontDescriptorCopyAttributes](#) (page 6)

Returns the attributes dictionary of the font descriptor.

[CTFontDescriptorCopyAttribute](#) (page 6)

Returns the value associated with an arbitrary attribute.

[CTFontDescriptorCopyLocalizedAttribute](#) (page 7)

Returns a localized value for the requested attribute, if available.

Getting the Font Descriptor Type

[CTFontDescriptorGetTypeID](#) (page 11)

Returns the type identifier for Core Text font descriptor references.

Functions

CTFontDescriptorCopyAttribute

Returns the value associated with an arbitrary attribute.

```
CTypeRef CTFontDescriptorCopyAttribute (
    CTFontDescriptorRef descriptor,
    CFStringRef attribute
);
```

Parameters

descriptor

The font descriptor.

attribute

The requested attribute.

Return Value

A retained reference to an arbitrary attribute, or NULL if the requested attribute is not present.

Discussion

Refer to [“Font Attributes”](#) (page 12) for documentation explaining how each attribute is packaged as a CType object.

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

CTFontDescriptorCopyAttributes

Returns the attributes dictionary of the font descriptor.

```
CFDictionaryRef CTFontDescriptorCopyAttributes (
    CTFontDescriptorRef descriptor
);
```

Parameters*descriptor*

The font descriptor.

Return Value

The font descriptor attributes dictionary. This dictionary contains the minimum number of attributes to specify fully this particular font descriptor.

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

CTFontDescriptorCopyLocalizedAttribute

Returns a localized value for the requested attribute, if available.

```
CTypeRef CTFontDescriptorCopyLocalizedAttribute (
    CTFontDescriptorRef descriptor,
    CFStringRef attribute,
    CFStringRef *language
);
```

Parameters*descriptor*

The font descriptor.

attribute

The requested font attribute.

language

On output, contains a reference to the matched language. The language identifier will conform to the RFC 3066bis standard.

Return Value

A retained reference to a localized attribute based on the global language list.

DiscussionThis function passes back the matched language in *language*. If localization is not possible for the attribute, the behavior matches the value returned from [CTFontDescriptorCopyAttribute](#) (page 6). Generally, localization of attributes is applicable to name attributes of only a normalized font descriptor.**Availability**

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

CTFontDescriptorCreateCopyWithAttributes

Creates a copy of the original font descriptor with new attributes.

```
CTFontDescriptorRef CTFontDescriptorCreateCopyWithAttributes (
    CTFontDescriptorRef original,
    CFDictionaryRef attributes
);
```

Parameters

original

The original font descriptor.

attributes

A dictionary containing arbitrary attributes.

Return Value

A new copy of the original font descriptor with attributes augmented by those specified. If there are conflicts between attributes, the new attributes replace existing ones.

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

CTFontDescriptorCreateCopyWithFeature

Copies a font descriptor with new feature settings.

```
CTFontDescriptorRef CTFontDescriptorCreateCopyWithFeature (
    CTFontDescriptorRef original,
    CFNumberRef featureTypeIdentifier,
    CFNumberRef featureSelectorIdentifier
);
```

Parameters

original

The original font descriptor.

featureTypeIdentifier

The feature type identifier.

featureSelectorIdentifier

The feature selector identifier.

Return Value

A copy of the original font descriptor modified with the given feature settings.

Discussion

This is a convenience method to toggle more easily the state of individual features.

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

CTFontDescriptorCreateCopyWithVariation

Creates a copy of the original font descriptor with a new variation instance.

```
CTFontDescriptorRef CTFontDescriptorCreateCopyWithVariation (
    CTFontDescriptorRef original,
    CFNumberRef variationIdentifier,
    CGFloat variationValue
);
```

Parameters

original

The original font descriptor.

variationIdentifier

The variation axis identifier. This is the four-character code of the variation axis as a CFNumber object.

variationValue

The value corresponding with the variation instance.

Return Value

A copy of the original font descriptor with a new variation instance.

Discussion

This is a convenience method for easily creating new variation font instances.

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

CTFontDescriptorCreateMatchingFontDescriptor

Returns the single preferred matching font descriptor based on the original descriptor and system precedence.

```
CTFontDescriptorRef CTFontDescriptorCreateMatchingFontDescriptor (
    CTFontDescriptorRef descriptor,
    CFSetRef mandatoryAttributes
);
```

Parameters

descriptor

The original font descriptor.

mandatoryAttributes

A set of attribute keys which must be identically matched in any returned font descriptors. May be NULL.

Return Value

A retained, normalized font descriptor matching the attributes present in *descriptor*.

Discussion

The original descriptor may be returned in normalized form. The caller is responsible for releasing the result. In the context of font descriptors, *normalized* infers that the input values were matched up with actual existing fonts, and the descriptors for those existing fonts are the returned normalized descriptors.

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

CTFontDescriptorCreateMatchingFontDescriptors

Returns an array of normalized font descriptors matching the provided descriptor.

```
CFArrayRef CTFontDescriptorCreateMatchingFontDescriptors (
    CTFontDescriptorRef descriptor,
    CFSetRef mandatoryAttributes
);
```

Parameters

descriptor

The font descriptor.

mandatoryAttributes

A set of attribute keys that must be identically matched in any returned font descriptors. May be NULL.

Return Value

A retained array of normalized font descriptors matching the attributes present in *descriptor*.

Discussion

If *descriptor* itself is normalized, then the array will contain only one item: the original descriptor. In the context of font descriptors, *normalized* infers that the input values were matched up with actual existing fonts, and the descriptors for those existing fonts are the returned normalized descriptors.

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

CTFontDescriptorCreateWithAttributes

Creates a new font descriptor reference from a dictionary of attributes.

```
CTFontDescriptorRef CTFontDescriptorCreateWithAttributes (
    CFDictionaryRef attributes
);
```

Parameters

attributes

A dictionary containing arbitrary attributes.

Return Value

A new font descriptor with the attributes specified.

Discussion

The provided attribute dictionary can contain arbitrary attributes that are preserved; however, unrecognized attributes are ignored on font creation and may not be preserved over the round trip from descriptor to font and back to descriptor.

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

CTFontDescriptorCreateWithNameAndSize

Creates a new font descriptor with the provided PostScript name and size.

```
CTFontDescriptorRef CTFontDescriptorCreateWithNameAndSize (
    CFStringRef name,
    CGFloat size
);
```

Parameters

name

The PostScript name to be used for the font descriptor as a `CFStringRef` object.

size

The point size. If 0.0, the font size attribute (`kCTFontSizeAttribute` (page 13)) is omitted from the returned font descriptor.

Return Value

A new font descriptor reference with the given PostScript name and point size.

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

CTFontDescriptorGetTypeID

Returns the type identifier for Core Text font descriptor references.

```
CTTypeID CTFontDescriptorGetTypeID (
    void
);
```

Return Value

The identifier for the `CTFontDescriptor` opaque type.

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

Data Types

CTFontDescriptorRef

A reference to a CTFontDescriptor object.

```
typedef const struct __CTFontDescriptor *CTFontDescriptorRef;
```

Availability

Available in Mac OS X v10.5 and later.

Declared In

CTFontDescriptor.h

Constants

Font Attributes

Font Attribute Constants

These constants are keys for accessing font attributes from a font descriptor.

```
const CFStringRef kCTFontNameAttribute;
const CFStringRef kCTFontDisplayNameAttribute;
const CFStringRef kCTFontFamilyNameAttribute;
const CFStringRef kCTFontStyleNameAttribute;
const CFStringRef kCTFontTraitsAttribute;
const CFStringRef kCTFontVariationAttribute;
const CFStringRef kCTFontSizeAttribute;
const CFStringRef kCTFontMatrixAttribute;
const CFStringRef kCTFontCascadeListAttribute;
const CFStringRef kCTFontCharacterSetAttribute;
const CFStringRef kCTFontLanguagesAttribute;
const CFStringRef kCTFontBaselineAdjustAttribute;
const CFStringRef kCTFontMacintoshEncodingsAttribute;
const CFStringRef kCTFontFeaturesAttribute;
const CFStringRef kCTFontFeatureSettingsAttribute;
const CFStringRef kCTFontFixedAdvanceAttribute;
const CFStringRef kCTFontOrientationAttribute;
```

Constants

kCTFontNameAttribute

Key for accessing the PostScript name from the font descriptor. The value associated with this key is a CFStringRef object. If the value is unspecified, it defaults to Helvetica, and if that font is unavailable, it falls back to the global font cascade list.

Available in Mac OS X v10.5 and later.

Declared in CTFontDescriptor.h.

`kCTFontDisplayNameAttribute`

Key for accessing the name used to display the font. Most commonly this is the full name. The value associated with this key is a `CFStringRef` object. If the value is unspecified, it defaults to `Helvetica`, and if that font is unavailable, it falls back to the global font cascade list.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontFamilyNameAttribute`

Key for accessing the font family name from the font descriptor. The value associated with this key is a `CFStringRef` object.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontStyleNameAttribute`

Key for accessing the style name of the font. This name represents the designer's description of the font's style. The value associated with this key is a `CFStringRef` object.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontTraitsAttribute`

Key for accessing the dictionary of font traits for stylistic information. See [“Font Traits”](#) (page 16) for the list of font traits. The value associated with this key is a `CFDictionaryRef` object.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontVariationAttribute`

Key to obtain the font variation dictionary instance as a `CFDictionaryRef` object. If specified in a font descriptor, fonts with the specified axes are primary match candidates; if no such fonts exist, this attribute is ignored.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontSizeAttribute`

Key to obtain or specify the font point size. Creating a font with this unspecified will default to a point size of 12.0. The value for this key is represented as a `CFNumberRef` object.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontMatrixAttribute`

Key to specify the font transformation matrix when creating a font. If unspecified it defaults to the unit matrix. The value for this key is a `CFDataRef` object containing a `CGAffineTransform` object.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontCascadeListAttribute`

Key to specify or obtain the cascade list used for a font reference. The cascade list is a `CFArrayRef` object containing `CTFontDescriptorRef` elements. If unspecified, the global cascade list is used.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontCharacterSetAttribute`

Key to specify or obtain the Unicode character coverage set for a font reference. The value for this key is a `CFCharacterSetRef` object. If specified, this attribute can be used to restrict the font to a subset of its actual character set. If unspecified, this attribute is ignored and the actual character set is used.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontLanguagesAttribute`

Key to specify or obtain a list of covered languages for a font reference. The value for this key is a `CFArrayRef` object containing `CFStringRef` elements. If specified, this attribute restricts the search to matching fonts that support the specified languages. The language identifier string should conform to the RFC 3066bis standard. If unspecified, this attribute is ignored.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontBaselineAdjustAttribute`

Key to specify or obtain the baseline adjustment for a font reference. This is primarily used when defining font descriptors for a cascade list to keep the baseline of all fonts even. The value associated with this is a float represented as a `CFNumberRef` object.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontMacintoshEncodingsAttribute`

Key to specify or obtain the Macintosh encodings for a font reference. The value associated with this key is a `CFNumberRef` object containing a bit field of the Macintosh encodings. This attribute is provided for legacy compatibility.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontFeaturesAttribute`

Key to specify or obtain the font features for a font reference. The value associated with this key is a `CFArrayRef` object containing font feature dictionaries. This feature list contains the feature information from the 'feat' table of the font. For more information, see `CTFontCopyFeatures`.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontFeatureSettingsAttribute`

Key to specify or obtain the font features settings for a font reference. The value associated with this key is a `CFArrayRef` object containing font feature-setting dictionaries. A feature-setting dictionary contains a tuple of a `kCTFontFeatureTypeIdentifierKey` key-value pair and a `kCTFontFeatureSelectorIdentifierKey` key-value pair. Each setting dictionary indicates which setting should be turned on. In the case of duplicate or conflicting setting, the last setting in the list takes precedence. It is the caller's responsibility to handle exclusive and nonexclusive settings as necessary.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontFixedAdvanceAttribute`

Key to specify a fixed advance to be used for a font reference. If present and specified, this attribute is used to specify a constant advance to override any font values. The value associated with this key is a float represented as a `CFNumberRef` object.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontOrientationAttribute`

Key to specify a particular orientation for the glyphs of the font. The value associated with this key is an integer represented as a `CFNumberRef` object. If you want to receive vertical metrics from a font for vertical rendering, specify `kCTFontVerticalOrientation` (page 15). If unspecified, the font uses its native orientation.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

Declared In

`CTFontDescriptor.h`

Font Orientation Constants

Specifies the intended rendering orientation of the font for obtaining glyph metrics.

```
enum {
    kCTFontDefaultOrientation = 0,
    kCTFontHorizontalOrientation = 1,
    kCTFontVerticalOrientation = 2
};
typedef uint32_t CTFontOrientation;
```

Constants`kCTFontDefaultOrientation`

The native orientation of the font.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontHorizontalOrientation`

Specifies horizontal orientation.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

`kCTFontVerticalOrientation`

Specifies vertical orientation.

Available in Mac OS X v10.5 and later.

Declared in `CTFontDescriptor.h`.

Declared In

`CTFontDescriptor.h`

Font Traits

Font Trait Constants

These constants are keys for accessing font traits from a font descriptor.

```
const CFStringRef kCTFontSymbolicTrait;
const CFStringRef kCTFontWeightTrait;
const CFStringRef kCTFontWidthTrait;
const CFStringRef kCTFontSlantTrait;
```

Constants

`kCTFontSymbolicTrait`

Key to access the symbolic traits value from the font traits dictionary. The value is returned as a `CFNumberRef` object.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontWeightTrait`

Key to access the normalized weight trait from the font traits dictionary. The value returned is a `CFNumberRef` representing a float value between `-1.0` and `1.0` for normalized weight. The value of `0.0` corresponds to the regular or medium font weight.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontWidthTrait`

Key to access the normalized proportion (width condense or expand) trait from the font traits dictionary. This value corresponds to the relative interglyph spacing for a given font. The value returned is a `CFNumberRef` object representing a float between `-1.0` and `1.0`. The value of `0.0` corresponds to regular glyph spacing, and negative values represent condensed glyph spacing.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontSlantTrait`

Key to access the normalized slant angle from the font traits dictionary. The value returned is a `CFNumberRef` object representing a float value between `-1.0` and `1.0` for normalized slant angle. The value of `0.0` corresponds to 0 degrees clockwise rotation from the vertical and `1.0` corresponds to 30 degrees clockwise rotation.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

Declared In

`CTFontTraits.h`

Font Class Mask Shift Constants

These constants represent the font class mask shift.


```
enum { kCTFontClassMaskShift = 28};
```

Constants

`kCTFontClassMaskShift`

Value used to shift the font class to the uppermost four bits of the symbolic traits

Declared In

`CTFontTraits.h`

Font Symbolic Traits Constants

These constants represent the symbolic representation of stylistic font attributes.

```
enum {
kCTFontItalicTrait = (1 << 0),
kCTFontBoldTrait = (1 << 1),
kCTFontExpandedTrait = (1 << 5),
kCTFontCondensedTrait = (1 << 6),
kCTFontMonoSpaceTrait = (1 << 10),
kCTFontVerticalTrait = (1 << 11),
kCTFontUIOptimizedTrait = (1 << 12),
kCTFontClassMaskTrait = (15 << kCTFontClassMaskShift)
};
typedef uint32_t CTFontSymbolicTraits;
```

Constants

`kCTFontItalicTrait`

The font typestyle is italic. Additional detail is available via [kCTFontSlantTrait](#) (page 16).

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontBoldTrait`

The font typestyle is boldface. Additional detail is available via [kCTFontWeightTrait](#) (page 16).

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontExpandedTrait`

The font typestyle is expanded. Expanded and condensed traits are mutually exclusive.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontCondensedTrait`

The font typestyle is condensed. Expanded and condensed traits are mutually exclusive. Additional detail is available via [kCTFontWidthTrait](#) (page 16).

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontMonoSpaceTrait`

The font uses fixed-pitch glyphs if available. The font may have multiple glyph advances (many CJK glyphs contain two spaces).

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontVerticalTrait`

The font uses vertical glyph variants and metrics.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontUIOptimizedTrait`

The font synthesizes appropriate attributes for user interface rendering, such as control titles, if necessary.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontClassMaskTrait`

Mask for the font class.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

Discussion

`CTFontSymbolicTraits` symbolically describes stylistic aspects of a font. The upper 16 bits are used to describe appearance of the font, whereas the lower 16 bits are for typeface information. The font appearance information represented by the upper 16 bits can be used for stylistic font matching.

Declared In

`CTFontTraits.h`

Font Stylistic Class Constants

These constants represent the stylistic class values of the font.

```
enum {
    kCTFontUnknownClass = (0 << kCTFontClassMaskShift),
    kCTFontOldStyleSerifsClass = (1 << kCTFontClassMaskShift),
    kCTFontTransitionalSerifsClass = (2 << kCTFontClassMaskShift),
    kCTFontModernSerifsClass = (3 << kCTFontClassMaskShift),
    kCTFontClarendonSerifsClass = (4 << kCTFontClassMaskShift),
    kCTFontSlabSerifsClass = (5 << kCTFontClassMaskShift),
    kCTFontFreeformSerifsClass = (7 << kCTFontClassMaskShift),
    kCTFontSansSerifClass = (8 << kCTFontClassMaskShift),
    kCTFontOrnamentalsClass = (9 << kCTFontClassMaskShift),
    kCTFontScriptsClass = (10 << kCTFontClassMaskShift),
    kCTFontSymbolicClass = (12 << kCTFontClassMaskShift)
};
typedef uint32_t CTFontStylisticClass;
```

Constants

`kCTFontUnknownClass`

The font has no design classification.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontOldStyleSerifsClass`

The font's style is based on the Latin printing style of the 15th to 17th century.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontTransitionalSerifsClass`

The font's style is based on the Latin printing style of the 18th to 19th century.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontModernSerifsClass`

The font's style is based on the Latin printing style of the 20th century.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontClarendonSerifsClass`

The font's style is a variation of the Oldstyle Serifs and the Transitional Serifs.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontSlabSerifsClass`

The font's style is characterized by serifs with a square transition between the strokes and the serifs (no brackets).

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontFreeformSerifsClass`

The font's style includes serifs, but it expresses a design freedom that does not generally fit within the other serif design classifications.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontSansSerifClass`

The font's style includes most basic letter forms (excluding Scripts and Ornaments) that do not have serifs on the strokes.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontOrnamentsClass`

The font's style includes highly decorated or stylized character shapes such as those typically used in headlines.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontScriptsClass`

The font's style is among those typefaces designed to simulate handwriting.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

`kCTFontSymbolicClass`

The font's style is generally design independent, making it suitable for special characters (icons, dingbats, technical symbols, and so on) that may be used equally well with any font.

Available in Mac OS X v10.5 and later.

Declared in `CTFontTraits.h`.

Discussion

`CTFontStylisticClass` classifies certain stylistic qualities of the font. These values correspond closely to the font class values in the OpenType OS/2 table. The class values are bundled in the upper four bits of the “[Font Symbolic Traits Constants](#)” (page 17) and can be obtained via `kCTFontClassMaskTrait` (page 18).

Declared In

`CTFontTraits.h`

Document Revision History

This table describes the changes to *CTFontDescriptor Reference*.

Date	Notes
2008-10-15	Corrected description of kCTFontStyleNameAttribute.
2007-05-30	New document that describes the Core Text opaque type used to represent a font descriptor.

REVISION HISTORY

Document Revision History

Index

C

CTFontDescriptorCopyAttribute **function 6**
CTFontDescriptorCopyAttributes **function 6**
CTFontDescriptorCopyLocalizedAttribute **function 7**
CTFontDescriptorCreateCopyWithAttributes **function 8**
CTFontDescriptorCreateCopyWithFeature **function 8**
CTFontDescriptorCreateCopyWithVariation **function 9**
CTFontDescriptorCreateMatchingFontDescriptor **function 9**
CTFontDescriptorCreateMatchingFontDescriptors **function 10**
CTFontDescriptorCreateWithAttributes **function 10**
CTFontDescriptorCreateWithNameAndSize **function 11**
CTFontDescriptorGetTypeID **function 11**
CTFontDescriptorRef **data type 12**

F

Font Attribute Constants **12**
Font Class Mask Shift Constants **16**
Font Orientation Constants **15**
Font Stylistic Class Constants **18**
Font Symbolic Traits Constants **17**
Font Trait Constants **16**

K

kCTFontBaselineAdjustAttribute **constant 14**
kCTFontBoldTrait **constant 17**
kCTFontCascadeListAttribute **constant 13**
kCTFontCharacterSetAttribute **constant 14**

kCTFontClarendonSerifsClass **constant 19**
kCTFontClassMaskShift **constant 17**
kCTFontClassMaskTrait **constant 18**
kCTFontCondensedTrait **constant 17**
kCTFontDefaultOrientation **constant 15**
kCTFontDisplayNameAttribute **constant 13**
kCTFontExpandedTrait **constant 17**
kCTFontFamilyNameAttribute **constant 13**
kCTFontFeaturesAttribute **constant 14**
kCTFontFeatureSettingsAttribute **constant 14**
kCTFontFixedAdvanceAttribute **constant 15**
kCTFontFreeformSerifsClass **constant 19**
kCTFontHorizontalOrientation **constant 15**
kCTFontItalicTrait **constant 17**
kCTFontLanguagesAttribute **constant 14**
kCTFontMacintoshEncodingsAttribute **constant 14**
kCTFontMatrixAttribute **constant 13**
kCTFontModernSerifsClass **constant 19**
kCTFontMonoSpaceTrait **constant 17**
kCTFontNameAttribute **constant 12**
kCTFontOldStyleSerifsClass **constant 18**
kCTFontOrientationAttribute **constant 15**
kCTFontOrnamentalsClass **constant 19**
kCTFontSansSerifClass **constant 19**
kCTFontScriptsClass **constant 19**
kCTFontSizeAttribute **constant 13**
kCTFontSlabSerifsClass **constant 19**
kCTFontSlantTrait **constant 16**
kCTFontStyleNameAttribute **constant 13**
kCTFontSymbolicClass **constant 19**
kCTFontSymbolicTrait **constant 16**
kCTFontTraitsAttribute **constant 13**
kCTFontTransitionalSerifsClass **constant 19**
kCTFontUIOptimizedTrait **constant 18**
kCTFontUnknownClass **constant 18**
kCTFontVariationAttribute **constant 13**
kCTFontVerticalOrientation **constant 15**
kCTFontVerticalTrait **constant 18**
kCTFontWeightTrait **constant 16**
kCTFontWidthTrait **constant 16**