

[Application Services](#) / Speech Synthesis Manager

API Collection

Speech Synthesis Manager

Overview

The Speech Synthesis Manager, formerly called the Speech Manager, is the part of macOS that provides a standardized method for Mac apps to generate synthesized speech. For example, you may want your application to incorporate the capability to speak its dialog box messages to the user. A word-processing application might use the Speech Synthesis Manager to implement a command that speaks a selected section of a document to the user. Because sound samples can take up large amounts of room on disk, using text in place of sampled sound is extremely efficient. For example, a multimedia application might use the Speech Synthesis Manager to provide a narration of a QuickTime movie instead of including sampled-sound data on a movie track.

OS X v10.5 introduces native support for performing speech synthesis tasks using Core Foundation-based objects, such as speaking text represented as `CFString` objects and managing speech channel properties using a `CFDictionary`-based property dictionary. You should begin using these Core Foundation-based programming interfaces as soon as it's convenient, because future synthesizers will accept Core Foundation strings and data structures directly through the speech synthesis framework. In the meantime, existing buffer-based clients and synthesizers will continue to work as before, with strings and other data structures getting automatically converted as necessary.

Gestalt Constants

You can check for version and feature availability information by using the Speech Synthesis Manager selectors defined in the [Gestalt Manager](#).

Topics

Changing Speech Attributes

~~func SetSpeechProperty(SpeechChannel, CFString, CFTypeRef?) -> OSErr~~

Sets the value of the specified speech-channel property.

Deprecated

~~func SetSpeechPitch(SpeechChannel, Fixed) -> OSErr~~

Sets the speech pitch on a designated speech channel.

Deprecated

~~func SetSpeechRate(SpeechChannel, Fixed) -> OSErr~~

Sets the speech rate of a designated speech channel.

Deprecated

Converting Text To Phonemes

~~func CopyPhonemesFromText(SpeechChannel, CFString, UnsafeMutablePointer<CFString?>) -> OSErr~~

Converts the specified text string into its equivalent phonemic representation.

Deprecated

Installing a Pronunciation Dictionary

~~func UseSpeechDictionary(SpeechChannel, CFDictionary) -> OSErr~~

Registers a speech dictionary with a speech channel.

Deprecated

Managing Speech Channels

~~func DisposeSpeechChannel(SpeechChannel) -> OSErr~~

Disposes of an existing speech channel.

Deprecated

~~func NewSpeechChannel(UnsafeMutablePointer<VoiceSpec?>, UnsafeMutablePointer<SpeechChannel?>) -> OSErr~~

Creates a new speech channel.

Deprecated

Obtaining Information About Speech and Speech Channels

~~func CopySpeechProperty(SpeechChannel, CFString, UnsafeMutablePointer<CTypeRef?>) -> OSErr~~

Gets the value associated with the specified property of a speech channel.

Deprecated

~~func GetSpeechPitch(SpeechChannel, UnsafeMutablePointer<Fixed>) -> OSErr~~

Gets a speech channel's current speech pitch.

Deprecated

~~func GetSpeechRate(SpeechChannel, UnsafeMutablePointer<Fixed>) -> OSErr~~

Gets a speech channel's current speech rate.

Deprecated

~~func SpeechBusy() -> Int16~~

Determines whether any channels of speech are currently synthesizing speech.

Deprecated

~~func SpeechBusySystemWide() -> Int16~~

Determines if any speech is currently being synthesized in your application or elsewhere on the computer.

Deprecated

~~func SpeechManagerVersion() -> NumVersion~~

Determines the current version of the Speech Synthesis Manager installed in the system.

Deprecated

Getting Information About Voices

~~func CountVoices(UnsafeMutablePointer<Int16>) -> OSErr~~

Determines how many voices are available.

Deprecated

~~func GetIndVoice(Int16, UnsafeMutablePointer<VoiceSpec>) -> OSErr~~

Gets a voice specification structure for a voice by passing an index to the GetIndVoice function.

Deprecated

```
func GetVoiceDescription(UnsafePointer<VoiceSpec>?, UnsafeMutablePointer<VoiceDescription>?, Int) -> OSErr
```

Gets a description of a voice by using the GetVoiceDescription function.

Deprecated

```
func GetVoiceInfo(UnsafePointer<VoiceSpec>?, OSType, UnsafeMutableRawPointer) -> OSErr
```

Gets the same information about a voice that the GetVoiceDescription function provides, or to determine in which file and resource a voice is stored.

Deprecated

```
func MakeVoiceSpec(OSType, OSType, UnsafeMutablePointer<VoiceSpec>) -> OSErr
```

Sets the fields of a voice specification structure.

Deprecated

Starting, Stopping, and Pausing Speech

```
func ContinueSpeech(SpeechChannel) -> OSErr
```

Resumes speech paused by the PauseSpeechAt function.

Deprecated

```
func PauseSpeechAt(SpeechChannel, Int32) -> OSErr
```

Pauses speech on a speech channel.

Deprecated

```
func SpeakCFString(SpeechChannel, CFString, CFDictionary?) -> OSErr
```

Begins speaking a string represented as a CFString object.

Deprecated

```
func StopSpeech(SpeechChannel) -> OSErr
```

Terminates speech immediately on the specified channel.

Deprecated

```
func StopSpeechAt(SpeechChannel, Int32) -> OSErr
```

Terminates speech delivery on a specified channel either immediately or at the end of the current word or sentence.

Deprecated

Registering and Unregistering Synthesizers and Voices

~~func SpeechSynthesisRegisterModuleURL(CFURL) -> OSErr~~

Registers and makes available a speech synthesizer or voice.

Deprecated

~~func SpeechSynthesisUnregisterModuleURL(CFURL) -> OSErr~~

Unregisters a registered speech synthesizer or voice.

Deprecated

Callbacks

`typedef SpeechDoneProcPtr`

Defines a pointer to a speech-done callback function which is called when the Speech Synthesis Manager finishes speaking a buffer of text.

`typedef SpeechErrorProcPtr`

Defines a pointer to an error callback function that handles syntax errors within commands embedded in a text buffer being processed by the Speech Synthesis Manager.

`typedef SpeechErrorCFProcPtr`

Defines a pointer to an error callback function that handles syntax errors within commands embedded in a CFString object being processed by the Speech Synthesis Manager.

`typedef SpeechPhonemeProcPtr`

Defines a pointer to a phoneme callback function that is called by the Speech Synthesis Manager before it pronounces a phoneme.

`typedef SpeechSyncProcPtr`

Defines a pointer to a synchronization callback function that is called when the Speech Synthesis Manager encounters a synchronization command embedded in a text buffer.

`typedef SpeechTextDoneProcPtr`

Defines a pointer to a text-done callback function that is called when the Speech Synthesis Manager has finished processing a buffer of text.

`typedef SpeechWordProcPtr`

Defines a pointer to a word callback function that is called by the Speech Synthesis Manager before it pronounces a word.

`typedef SpeechWordCFProcPtr`

Defines a pointer to a Core Foundation-based word callback function that is called by the Speech Synthesis Manager before it pronounces a word.

Data Types

`struct DelimiterInfo`

Defines a delimiter information structure.

`struct PhonemeDescriptor`

Defines a phoneme descriptor structure.

`struct PhonemeInfo`

Defines a structure that stores information about a phoneme.

`struct SpeechChannelRecord`

Represents a speech channel.

`typedef SpeechChannel`

Defines a pointer to a speech channel record.

`typedef SpeechDoneUPP`

Defines a universal procedure pointer (UPP) to a speech-done callback function.

`struct SpeechErrorInfo`

Defines a speech error information structure.

`typedef SpeechErrorUPP`

Defines a universal procedure pointer (UPP) to an error callback function.

`typedef SpeechPhonemeUPP`

Defines a universal procedure pointer (UPP) to a phoneme callback function.

`struct SpeechStatusInfo`

Defines a speech status information structure, which stores information about the status of a speech channel.

`typedef SpeechSyncUPP`

Defines a universal procedure pointer (UPP) to a synchronization callback function.

`typedef SpeechTextDoneUPP`

Defines a universal procedure pointer (UPP) to a text-done callback function.

`struct SpeechVersionInfo`

Defines a speech version information structure.

`typedef` SpeechWordUPP

Defines a universal procedure pointer (UPP) to a word callback function.

`struct` SpeechXtndData

Defines a speech extension data structure.

`struct` VoiceDescription

Defines a voice description structure.

`struct` VoiceFileInfo

Defines a voice file information structure.

`struct` VoiceSpec

Defines a voice specification structure.

Constants

≡ Control Flags Constants

Flags that indicate which synthesizer features are active.

≡ Gender Constants

Constants that indicate the gender of the individual represented by a voice.

≡ Audio Unit Constants

Constants that identify values in a speech synthesis audio unit.

≡ Stop Speech Locations

Locations that indicate where speech should be paused or stopped.

≡ Speech Synthesis Manager Operating System Types

The OSType definitions used by the Speech Synthesis Manager.

≡ Speech-Channel Modes

The available text-processing and number-processing modes for a speech channel.

≡ Speech-Channel Modes for Core Foundation-based Functions

The available text-processing and number-processing modes for a speech channel.

≡ Voice Information Selectors

The types of voice data that can be requested by the GetVoiceInfo function.

≡ Speech-Channel Information Constants

Selectors that can be passed to the `GetSpeechInfo` or `SetSpeechInfo` functions.

≡ Phoneme Generation Options

Flags that specify options for the generation of phonetic output.

≡ Speech-Channel Properties

Properties used with `CopySpeechProperty(: : :)` or `SetSpeechProperty(: : :)` to get or set the characteristics of a speech channel.

≡ Synthesizer Option Keys

Keys used to specify synthesizer options.

≡ Speech Status Keys

Keys used with the `kSpeechStatusProperty` property to specify the status of the speech channel.

≡ Speech Error Keys

Keys used with the `kSpeechErrorsProperty` property to describe errors encountered during speech processing and production.

≡ Speech Synthesizer Information Keys

Keys used with the `kSpeechSynthesizerInfoProperty` property to get information about the synthesizer.

≡ Phoneme Symbols Keys

Keys used with the `kSpeechPhonemeSymbolsProperty` property to provide information about the phoneme being processed.

≡ Current Voice Keys

Keys used with the `kSpeechCurrentVoiceProperty` property to specify information about the current voice.

≡ Command Delimiter Keys

Keys used with the `kSpeechCommandDelimiterProperty` property to specify information about the command delimiter strings.

≡ Speech Dictionary Keys

Keys used in a speech dictionary to override the synthesizer's default pronunciation of a word.

≡ Error Callback User-Information String

Specifies information about the text being synthesized when an error occurs.

Result Codes

The most common result codes returned by `Speech SynthesisManager` are listed below.

<code>var noSynthFound: Int</code>	Could not find the specified speech synthesizer
<code>var synthOpenFailed: Int</code>	Could not open another speech synthesizerchannel
<code>var synthNotReady: Int</code>	Speech synthesizer is still busy speaking
<code>var bufTooSmall: Int</code>	Output buffer is too small to hold result
<code>var voiceNotFound: Int</code>	Voice resource not found
<code>var incompatibleVoice: Int</code>	Specified voice cannot be used with synthesizer
<code>var badDictFormat: Int</code>	Pronunciation dictionary format error
<code>var badInputText: Int</code>	Raw phoneme text contains invalid characters

See Also

Managers

- ⋮ [Apple Event Manager](#)
- ⋮ [ColorSync Manager](#)