NSAppleScript Class Reference

Cocoa > Scripting & Automation

2007-07-10

Apple Inc.
© 2007 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, AppleScript, Cocoa, Mac, and Mac OS are trademarks of Apple Inc., registered in the United States and other countries.

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

NSAppleScript Class Reference 5

Overview 5
Adopted Protocols 6
Tasks 6
- Initializing a Script 6
- Getting Information About a Script 6
- Compiling and Executing a Script 6

Instance Methods 6
- compileAndReturnError: 6
- executeAndReturnError: 7
- executeAppleEvent:error: 7
- initWithContentsOfURL:error: 8
- initWithSource: 8
- isCompiled 9
- source 9

Constants 9
- Error Dictionary Keys 9

Document Revision History 11

Index 13

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.

NSAppleScript Class Reference

Inherits from	NSObject
Conforms to	NSCopying NSObject (NSObject)
Framework	/System/Library/Frameworks/Foundation.framework
Availability	Available in Mac OS X v10.2 and later.
Companion guide	Cocoa Scripting Guide
Declared in	NSAppleScript.h
Related sample code	AttachAScript

Overview

The `NSAppleScript` class provides the ability to load, compile, and execute scripts.

Important: You should access `NSAppleScript` only from the main thread.

This class provides applications with the ability to

- load a script from a URL or from a text string
- compile or execute a script or an individual Apple event
- obtain an `NSAppleEventDescriptor` containing the reply from an executed script or event
- obtain an attributed string for a compiled script, suitable for display in a script editor
- obtain various kinds of information about any errors that may occur

Important: `NSAppleScript` provides the `executeAppleEvent:error:` (page 7) method so that you can send an Apple event to invoke a handler in a script. (In an AppleScript script, a handler is the equivalent of a function.) However, you cannot use this method to send Apple events to other applications.

When you create an instance of `NSAppleScript` object, you can use a URL to specify a script that can be in either text or compiled form, or you can supply the script as a string. Should an error occur when compiling or executing the script, several of the methods return a dictionary containing error information. The keys for obtaining error information, such as `NSAppleScriptErrorMessage` (page 10), are described in the Constants section.

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.

See also NSAppleScript Additions in the Application Kit framework, which defines a method that returns the syntax-highlighted source code for a script.

Adopted Protocols

NSCopying
- `copyWithZone:`

Tasks

Initializing a Script

- `initWithContentsOfURL:error:` (page 8)
  Initializes a newly allocated script instance from the source identified by the passed URL.
- `initWithSource:` (page 8)
  Initializes a newly allocated script instance from the passed source.

Getting Information About a Script

- `isCompiled` (page 9)
  Returns a Boolean value that indicates whether the receiver's script has been compiled.
- `source` (page 9)
  Returns the script source for the receiver.

Compiling and Executing a Script

- `compileAndReturnError:` (page 6)
  Compiles the receiver, if it is not already compiled.
- `executeAndReturnError:` (page 7)
  Executes the receiver, compiling it first if it is not already compiled.
- `executeAppleEvent:error:` (page 7)
  Executes an Apple event in the context of the receiver, as a means of allowing the application to invoke a handler in the script.

Instance Methods

compileAndReturnError:

Compiles the receiver, if it is not already compiled.

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.

```
- (BOOL)compileAndReturnError:(NSDictionary **)errorInfo
```

Parameters

errorInfo

On return, if an error occurs, a pointer to an error information dictionary.

Return Value

`YES` for success or if the script was already compiled, `NO` otherwise.

Availability

Available in Mac OS X v10.2 and later.

Declared In

`NSAppleScript.h`

executeAndReturnError:

Executes the receiver, compiling it first if it is not already compiled.

```
- (NSAppleEventDescriptor *)executeAndReturnError:(NSDictionary **)errorInfo
```

Parameters

errorInfo

On return, if an error occurs, a pointer to an error information dictionary.

Return Value

The result of executing the event, or `nil` if an error occurs.

Discussion

Any changes to property values caused by executing the script do not persist.

Availability

Available in Mac OS X v10.2 and later.

Declared In

`NSAppleScript.h`

executeAppleEvent:error:

Executes an Apple event in the context of the receiver, as a means of allowing the application to invoke a handler in the script.

```
- (NSAppleEventDescriptor *)executeAppleEvent:(NSAppleEventDescriptor *)event
    error:(NSDictionary **)errorInfo
```

Parameters

event

The Apple event to execute.

errorInfo

On return, if an error occurs, a pointer to an error information dictionary.

Return Value

The result of executing the event, or `nil` if an error occurs.

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.

Discussion
Compiles the receiver before executing it if it is not already compiled.

Important: You cannot use this method to send Apple events to other applications.

Availability
Available in Mac OS X v10.2 and later.

Declared In
`NSAppleScript.h`

initWithContentsOfURL:error:

Initializes a newly allocated script instance from the source identified by the passed URL.

```
- (id)initWithContentsOfURL:(NSURL *)url error:(NSDictionary **)errorInfo
```

Parameters

url
A URL that locates a script, in either text or compiled form.

errorInfo
On return, if an error occurs, a pointer to an error information dictionary.

Return Value
The initialized script object, `nil` if an error occurs.

Discussion
This method is a designated initializer for `NSAppleScript`.

Availability
Available in Mac OS X v10.2 and later.

Declared In
`NSAppleScript.h`

initWithSource:

Initializes a newly allocated script instance from the passed source.

```
- (id)initWithSource:(NSString *)source
```

Parameters

source
A string containing the source code of a script.

Return Value
The initialized script object, `nil` if an error occurs.

Discussion
This method is a designated initializer for `NSAppleScript`.

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.

Availability
Available in Mac OS X v10.2 and later.

Declared In
`NSAppleScript.h`

isCompiled

Returns a Boolean value that indicates whether the receiver's script has been compiled.

```
- (BOOL)isCompiled
```

Return Value
`YES` if the receiver is already compiled, `NO` otherwise.

Availability
Available in Mac OS X v10.2 and later.

Declared In
`NSAppleScript.h`

source

Returns the script source for the receiver.

```
- (NSString *)source
```

Return Value
The script source code of the receiver if it is available, `nil` otherwise.

Discussion
It is possible for an `NSAppleScript` that has been instantiated with `initWithContentsOfURL:error:` (page 8) to be a script for which the source code is not available but is nonetheless executable.

Availability
Available in Mac OS X v10.2 and later.

Declared In
`NSAppleScript.h`

Constants

Error Dictionary Keys

If the result of `initWithContentsOfURL:error:` (page 8), `compileAndReturnError:` (page 6), `executeAndReturnError:` (page 7), or `executeAppleEvent:error:` (page 7), signals failure (`nil`, `NO`, `nil`, or `nil`, respectively), a pointer to an autoreleased dictionary is put at the location pointed to by the error parameter. The error info dictionary may contain entries that use any combination of the following keys, including no entries at all.

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.

```
extern NSString *NSAppleScriptErrorMessage;
extern NSString *NSAppleScriptErrorNumber;
extern NSString *NSAppleScriptErrorAppName;
extern NSString *NSAppleScriptErrorBriefMessage;
extern NSString *NSAppleScriptErrorRange;
```

Constants

`NSAppleScriptErrorMessage`

An `NSString` that supplies a detailed description of the error condition.

Available in Mac OS X v10.2 and later.

Declared in `NSAppleScript.h`.

`NSAppleScriptErrorNumber`

An `NSNumber` that specifies the error number.

Available in Mac OS X v10.2 and later.

Declared in `NSAppleScript.h`.

`NSAppleScriptErrorAppName`

An `NSString` that specifies the name of the application that generated the error.

Available in Mac OS X v10.2 and later.

Declared in `NSAppleScript.h`.

`NSAppleScriptErrorBriefMessage`

An `NSString` that provides a brief description of the error.

Available in Mac OS X v10.2 and later.

Declared in `NSAppleScript.h`.

`NSAppleScriptErrorRange`

An `NSValue` that specifies a range.

Available in Mac OS X v10.2 and later.

Declared in `NSAppleScript.h`.

Declared In

`NSAppleScript.h`

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.

Document Revision History

This table describes the changes to *NSAppleScript Class Reference*.

Date	Notes
2007-07-10	Added availability information for constants.
	For the `executeAndReturnError:` (page 7) method, noted that any changes to property values caused by executing the script do not persist.
2006-05-23	First publication of this content as a separate document.

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.

Document Revision History

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.

Index

C

`compileAndReturnError:` instance method 6

E

Error Dictionary Keys 9
`executeAndReturnError:` instance method 7
`executeAppleEvent:error:` instance method 7

I

`initWithContentsOfURL:error:` instance method 8
`initWithSource:` instance method 8
`isCompiled` instance method 9

N

`NSAppleScriptErrorAppName` constant 10
`NSAppleScriptErrorBriefMessage` constant 10
`NSAppleScriptErrorMessage` constant 10
`NSAppleScriptErrorNumber` constant 10
`NSAppleScriptErrorRange` constant 10

S

`source` instance method 9

2007-07-10 | © 2007 Apple Inc. All Rights Reserved.