

---

# WebFrame Class Reference

[Cocoa](#) > [User Experience](#)



2009-04-08



Apple Inc.  
© 2009 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, Cocoa, Mac, Mac OS, Objective-C, and Safari are trademarks of Apple Inc., registered in the United States and other countries.

WebScript is a trademark of Apple Inc.

Java and all Java-based trademarks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. 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

---

## WebFrame Class Reference 5

---

Overview	5
Tasks	6
Initializing Frames	6
Loading Content	6
Getting the Data Source	6
Getting Related Frames and Views	6
Finding Frames	7
Getting DOM Objects	7
Instance Methods	7
childFrames	7
dataSource	7
DOMDocument	8
findFrameNamed:	8
frameElement	9
frameView	9
globalContext	9
initWithName:webFrameView:webView:	10
loadAlternateHTMLString:baseURL:forUnreachableURL:	10
loadArchive:	11
loadData:MIMEType:textEncodingName:baseURL:	11
loadHTMLString:baseURL:	12
loadRequest:	12
name	13
parentFrame	13
provisionalDataSource	14
reload	14
stopLoading	15
webView	15
windowObject	15

---

## Document Revision History 17

---

## Index 19

---



# WebFrame Class Reference

---

<b>Inherits from</b>	NSObject
<b>Conforms to</b>	NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/WebKit.framework
<b>Availability</b>	Available in Mac OS X v10.2 with Safari 1.0 and later. Available in Mac OS X v10.2.7 and later.
<b>Companion guide</b>	WebKit Objective-C Programming Guide
<b>Declared in</b>	WebFrame.h
<b>Related sample code</b>	CarbonCocoaCoreImageTab HIView-NSView NewsReader

## Overview

A `WebFrame` object encapsulates the data displayed in a `WebFrameView` object. There is one `WebFrame` object per frame displayed in a `WebView`. An entire webpage is represented by a hierarchy of `WebFrame` objects in which the root object is called the **main frame**.

Each `WebFrame` also has a `WebDataSource` object that manages the loading of frame content. You use the [loadRequest:](#) (page 12) method to initiate an asynchronous client request which will create a provisional data source. The provisional data source will transition to a committed data source once any data has been received.

There are some special, predefined, frame names that you can use when referring to or finding a `WebFrame`. Some of the predefined frame names are: “\_self”, “\_current”, “\_parent”, and “\_top.” See [findFrameNamed:](#) (page 8) for a description of their meaning. Frame names may also be specified in the HTML source, or set by clients.

However, the group name is an arbitrary identifier used to group related frames. For example, JavaScript running in a frame can access any other frame in the same group. It's up to the application how it chooses to scope related frames.

## Tasks

### Initializing Frames

- [initWithName:webFrameView:webView:](#) (page 10)  
Initializes the receiver with a frame name, web frame view, and controlling web view.

### Loading Content

- [loadRequest:](#) (page 12)  
Connects to a given URL by initiating an asynchronous client request.
- [reload](#) (page 14)  
Reloads the initial request passed as an argument to [loadRequest:](#) (page 12).
- [stopLoading](#) (page 15)  
Stops any pending loads on the receiver's data source, and those of its children.
- [loadAlternateHTMLString:baseURL:forUnreachableURL:](#) (page 10)  
Loads alternate content for a frame whose URL is unreachable.
- [loadHTMLString:baseURL:](#) (page 12)  
Sets the main page contents and base URL.
- [loadData:MIMETYPE:textEncodingName:baseURL:](#) (page 11)  
Sets the main page contents, MIME type, content encoding, and base URL.
- [loadArchive:](#) (page 11)  
Loads an archive into the web frame.

### Getting the Data Source

- [dataSource](#) (page 7)  
Returns the committed data source.
- [provisionalDataSource](#) (page 14)  
Returns the provisional data source.

### Getting Related Frames and Views

- [parentFrame](#) (page 13)  
Returns the web frame's parent web frame.
- [childFrames](#) (page 7)  
Returns the frames of the web frame's immediate children.
- [frameView](#) (page 9)  
Returns the web frame's view object.
- [webView](#) (page 15)  
Returns the view object that manages the web frame.

## Finding Frames

- [findFrameNamed:](#) (page 8)  
Returns a web frame that matches the given name.
- [name](#) (page 13)  
Returns the web frame's name.

## Getting DOM Objects

- [DOMDocument](#) (page 8)  
Returns the web frame's DOM document.
- [frameElement](#) (page 9)  
Returns the web view's DOM frame element.
- [globalContext](#) (page 9)  
Returns the global JavaScript execution context for bridging between the WebKit and JavaScriptCore C API.
- [windowObject](#) (page 15)  
Returns the JavaScript window object.

## Instance Methods

### childFrames

Returns the frames of the web frame's immediate children.

- (NSArray \*)childFrames

#### Return Value

The web frame's immediate children. Each child web frame is an instance of `WebFrame` and corresponds to an HTML frameset or `iframe` element.

#### Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

#### See Also

- [parentFrame](#) (page 13)

#### Declared In

WebFrame.h

### dataSource

Returns the committed data source.

- (WebDataSource \*)dataSource

**Return Value**

The committed data source, or `nil` if the provisional data source is not done loading.

**Availability**

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

**See Also**

- [provisionalDataSource](#) (page 14)

**Related Sample Code**

CarbonCocoaCoreImageTab

**Declared In**

WebFrame.h

**DOMDocument**

Returns the web frame's DOM document.

```
- (DOMDocument *)DOMDocument
```

**Return Value**

The web frame's DOM document.

**Discussion**

Returns `nil` if the receiver doesn't have a DOM document; for example, if it's a standalone image.

**Availability**

Available in Mac OS X v10.3.9 and later.

**Declared In**

WebFrame.h

**findFrameNamed:**

Returns a web frame that matches the given name.

```
- (WebFrame *)findFrameNamed:(NSString *)name
```

**Parameters**

*name*

The name of a frame.

**Return Value**

For predefined names, returns the receiver if name is “\_self” or “\_current”; returns the receiver's parent frame if name is “\_parent”; and returns the main frame if name is “\_top”. Also returns the receiver if it is the main frame and name is either “\_parent” or “\_top.” For other names, this method returns the first frame that matches *name*. Returns `nil` if no match is found.

**Discussion**

This method searches the receiver and its descendents first, then the receiver's parent and its children moving up the hierarchy until a match is found. If no match is found in the receivers hierarchy, this method will search for a matching frame in other main frame hierarchies.



**Availability**

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

**See Also**

- [name](#) (page 13)

**Declared In**

WebFrame.h

## frameElement

Returns the web view's DOM frame element.

- (DOMHTML\*Element \*)frameElement

**Return Value**

The web view's DOM frame element. Returns `nil` if the receiver is the main frame.

**Discussion**

The returned object may be an instance of either `DOMHTMLFrameElement`, `DOMHTMLIFrameElement` or `DOMHTMLObjectElement`.

**Availability**

Available in Mac OS X v10.3.9 and later.

**Declared In**

WebFrame.h

## frameView

Returns the web frame's view object.

- (WebFrameView \*)frameView

**Return Value**

The web frame's view object.

**Availability**

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

**See Also**

- [webView](#) (page 15)

**Declared In**

WebFrame.h

## globalContext

Returns the global JavaScript execution context for bridging between the WebKit and JavaScriptCore C API.

```
- (JSGlobalContextRef)globalContext
```

**Return Value**

The global JavaScript execution context.

**Availability**

Available in Mac OS X v10.4.11 and later.

**Declared In**

WebFrame.h

**initWithName:webFrameView:webView:**

Initializes the receiver with a frame name, web frame view, and controlling web view.

```
- (id)initWithName:(NSString *)frameName webFrameView:(WebFrameView *)frameView
  webView:(WebView *)webView
```

**Parameters**

*frameName*

The frame name. Typically a custom name or `nil` (if none is specified). It would be inappropriate to use one of the predefined frame names described in [findFrameNamed:](#) (page 8) as they have special meanings.

*view*

The view that displays this web frame—the view associated with the receiver.

*webView*

The parent view that manages the main frame and its children.

**Return Value**

An initialized web frame.

**Discussion**

Normally, you do not invoke this method directly. `WebView` objects automatically create the main frame and subsequent children when new content is loaded. Send a [loadRequest:](#) (page 12) message to the main frame of a `WebView` to load web content.

This method is the designated initializer for the `WebFrame` class.

**Availability**

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

**Declared In**

WebFrame.h

**loadAlternateHTMLString:baseURL:forUnreachableURL:**

Loads alternate content for a frame whose URL is unreachable.

```
- (void)loadAlternateHTMLString:(NSString *)string baseURL:(NSURL *)URL
  forUnreachableURL:(NSURL *)unreachableURL
```

**Parameters***string*

The string to use as the main page for the document.

*URL*

A file that is used to resolve relative URLs within the document.

*unreachableURL*

The URL for the alternate page content.

**Discussion**

Use this method to display page-level loading errors in a web view. Typically, a `WebFrameLoadDelegate` or `WebPolicyDelegate` object invokes this method from these methods:

`webView:didFailProvisionalLoadWithError:forFrame:(WebFrameLoadDelegate)`,  
`webView:decidePolicyForMIMEType:request:frame:decisionListener:(WebPolicyDelegate)`,  
 or `webView:unableToImplementPolicyWithError:frame:(WebPolicyDelegate)`. If invoked from one of these methods, the back-forward list is maintained.

**Availability**

Available in Mac OS X v10.3.9 and later.

**Declared In**

WebFrame.h

**loadArchive:**

Loads an archive into the web frame.

```
- (void)loadArchive:(WebArchive *)archive
```

**Parameters***archive*

The archive to load.

**Availability**

Available in Mac OS X v10.3.9 and later.

**Declared In**

WebFrame.h

**loadData:MIMEType:textEncodingName:baseURL:**

Sets the main page contents, MIME type, content encoding, and base URL.

```
- (void)loadData:(NSData *)data MIMEType:(NSString *)MIMEType
  textEncodingName:(NSString *)encodingName baseURL:(NSURL *)URL
```

**Parameters***data*

The data to use for the main page of the document.

*MIMEType*

The MIME type of the data.

*encodingName*

The IANA encoding name (for example, “utf-8” or “utf-16”).

*URL*

A file that is used to resolve relative URLs within the document.

**Availability**

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

**See Also**

- [loadHTMLString:baseURL:](#) (page 12)

**Declared In**

WebFrame.h

## **loadHTMLString:baseURL:**

Sets the main page contents and base URL.

- (void)loadHTMLString:(NSString \*)*string* baseURL:(NSURL \*)*URL*

**Parameters**

*string*

The string to use as the main page for the document.

Since the string is treated as a webpage with UTF-8 encoding, the default encoding for any script elements referenced by the HTML is also UTF-8. To avoid this, include a character set attribute on the script element.

*URL*

A file that is used to resolve relative URLs within the document.

**Availability**

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

**See Also**

- [loadData:MIMETYPE:textEncodingName:baseURL:](#) (page 11)

**Declared In**

WebFrame.h

## **loadRequest:**

Connects to a given URL by initiating an asynchronous client request.

- (void)loadRequest:(NSURLRequest \*)*request*

**Parameters**

*request*

A client request.

### Discussion

Creates a provisional data source that will transition to a committed data source once any data has been received. Use the [dataSource](#) (page 7) method to check if a committed data source is available, and the [stopLoading](#) (page 15) method to stop the load. This method is typically invoked on the main frame.

### Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

### See Also

- [reload](#) (page 14)
- [stopLoading](#) (page 15)

### Related Sample Code

CarbonCocoaCoreImageTab

### Declared In

WebFrame.h

## name

Returns the web frame's name.

```
- (NSString *)name
```

### Return Value

The web frame's name.

### Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

### See Also

- [findFrameNamed:](#) (page 8)

### Declared In

WebFrame.h

## parentFrame

Returns the web frame's parent web frame.

```
- (WebFrame *)parentFrame
```

### Return Value

The parent web frame, or `nil` if it has none.

### Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

**See Also**

- [childFrames](#) (page 7)

**Declared In**

WebFrame.h

## provisionalDataSource

Returns the provisional data source.

- (WebDataSource \*)provisionalDataSource

**Return Value**

The provisional data source, or `nil` if either a load request is not in progress or a load request has completed.

**Discussion**

Use the [loadRequest:](#) (page 12) method to initiate an asynchronous client request, which creates a provisional data source. The provisional data source transitions to a committed data source once any data is received.

**Availability**

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

**See Also**

- [dataSource](#) (page 7)

**Related Sample Code**

CarbonCocoaCoreImageTab

**Declared In**

WebFrame.h

## reload

Reloads the initial request passed as an argument to [loadRequest:](#) (page 12).

- (void)reload

**Availability**

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

**See Also**

- [loadRequest:](#) (page 12)

- [stopLoading](#) (page 15)

**Declared In**

WebFrame.h

## stopLoading

Stops any pending loads on the receiver's data source, and those of its children.

- (void)stopLoading

### Discussion

This method does not change the state of the receiver—whatever content has been loaded is preserved.

### Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

### See Also

- [loadRequest:](#) (page 12)

- [reload](#) (page 14)

### Declared In

WebFrame.h

## webView

Returns the view object that manages the web frame.

- (WebView \*)webView

### Return Value

The view object that manages the entire hierarchy of web frame objects that contains the receiver.

### Availability

Available in Mac OS X v10.2 with Safari 1.0 and later.

Available in Mac OS X v10.2.7 and later.

### See Also

- [frameView](#) (page 9)

### Declared In

WebFrame.h

## windowObject

Returns the JavaScript window object.

- (WebScriptObject \*)windowObject

### Return Value

The JavaScript window object.

### Availability

Available in Mac OS X v10.4.11 and later.

### Declared In

WebFrame.h





# Document Revision History

---

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

Date	Notes
2009-04-08	Added separate parameter and return value descriptions.
2008-10-15	Minor edits throughout.
2008-06-19	Updated for Mac OS X v10.5.
2006-05-23	First publication of this content as a separate document.

## REVISION HISTORY

### Document Revision History

# Index

---

## C

---

childFrames [instance method 7](#)

## D

---

dataSource [instance method 7](#)  
DOMDocument [instance method 8](#)

## F

---

findFrameNamed: [instance method 8](#)  
frameElement [instance method 9](#)  
frameView [instance method 9](#)

## G

---

globalContext [instance method 9](#)

## I

---

initWithName:webView: [instance method 10](#)

## L

---

loadAlternateHTMLString:baseURL:forUnreachableURL: [instance method 10](#)  
loadArchive: [instance method 11](#)  
loadData:MIMETYPE:textEncodingName:baseURL: [instance method 11](#)  
loadHTMLString:baseURL: [instance method 12](#)  
loadRequest: [instance method 12](#)

## N

---

name [instance method 13](#)

## P

---

parentFrame [instance method 13](#)  
provisionalDataSource [instance method 14](#)

## R

---

reload [instance method 14](#)

## S

---

stopLoading [instance method 15](#)

## W

---

webView [instance method 15](#)  
windowObject [instance method 15](#)