

---

# NSClassDescription Class Reference

[Cocoa](#) > [AppleScript](#)



2006-05-23



Apple Inc.  
© 2006 Apple Computer, 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

---

## **NSClassDescription Class Reference 5**

---

Overview 5

Tasks 6

    Working with Class Descriptions 6

    Attribute Keys 6

    Relationship Keys 6

Class Methods 6

    classDescriptionForClass: 6

    invalidateClassDescriptionCache 7

    registerClassDescription:forClass: 7

Instance Methods 8

    attributeKeys 8

    inverseForRelationshipKey: 8

    toManyRelationshipKeys 9

    toOneRelationshipKeys 9

Notifications 10

    NSClassDescriptionNeededForClassNotification 10

---

## **Document Revision History 11**

---

## **Index 13**

---



# NSClassDescription Class Reference

---

<b>Inherits from</b>	NSObject
<b>Conforms to</b>	NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/Foundation.framework
<b>Availability</b>	Available in Mac OS X v10.0 and later.
<b>Declared in</b>	NSClassDescription.h
<b>Companion guides</b>	Cocoa Scripting Guide Key-Value Coding Programming Guide

## Overview

`NSClassDescription` is an abstract class that provides the interface for querying the relationships and properties of a class. Concrete subclasses of `NSClassDescription` provide the available attributes of objects of a particular class and the relationships between that class and other classes. Defining these relationships between classes allows for more intelligent and flexible manipulation of objects with key-value coding.

It is important to note that there are no class descriptions by default. To use `NSClassDescription` objects in your code you have to implement them for your model classes. For all concrete subclasses, you must provide implementations for all instance methods of `NSClassDescription`. (`NSClassDescription` provides only the implementation for the class methods that maintain the cache of registered class descriptions.) Once created, you must register a class description with the `NSClassDescription` method [registerClassDescription:forClass:](#) (page 7).

You can use the `NSString` objects in the arrays returned by methods such as [attributeKeys](#) (page 8) and [toManyRelationshipKeys](#) (page 9) to access—using key-value coding—the properties of an instance of the class to which a class description object corresponds. For more about attributes and relationships, see *Cocoa Fundamentals Guide*. For more about key-value coding, see *Key-Value Coding Programming Guide*.

`NSScriptClassDescription`, which is used to map the relationships between scriptable classes, is the only concrete subclass of `NSClassDescription` provided as part of the Cocoa framework.

## Tasks

### Working with Class Descriptions

- + `classDescriptionForClass:` (page 6)  
Returns the class description for a given class.
- + `invalidateClassDescriptionCache` (page 7)  
Removes all `NSClassDescription` objects from the cache.
- + `registerClassDescription:forClass:` (page 7)  
Registers an `NSClassDescription` object for a given class in the `NSClassDescription` cache.

### Attribute Keys

- `attributeKeys` (page 8)  
Overridden by subclasses to return the names of attributes of instances of the described class.

### Relationship Keys

- `inverseForRelationshipKey:` (page 8)  
Overridden by subclasses to return the name of the inverse relationship from a relationship specified by a given key.
- `toManyRelationshipKeys` (page 9)  
Overridden by subclasses to return the keys for the to-many relationship properties of instances of the described class.
- `toOneRelationshipKeys` (page 9)  
Overridden by subclasses to return the keys for the to-one relationship properties of instances of the described class.

## Class Methods

### `classDescriptionForClass:`

Returns the class description for a given class.

```
+ (NSClassDescription *)classDescriptionForClass:(Class)aClass
```

#### Parameters

*aClass*

The class for which to return a class description.

#### Return Value

The class description for *aClass*, or `nil` if a class description cannot be found.

**Discussion**

If a class description for *aClass* is not found, the method posts an `NSClassDescriptionNeededForClassNotification` on behalf of *aClass*, allowing an observer to register a class description. The method then checks for a class description again. Returns `nil` if a class description is still not found.

If you have an instance of the receiver's class, you can use the `NSObject` instance method `classDescription` instead.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

`NSClassDescription.h`

**invalidateClassDescriptionCache**

Removes all `NSClassDescription` objects from the cache.

```
+ (void)invalidateClassDescriptionCache
```

**Discussion**

You should rarely need to invoke this method. Use it whenever a registered `NSClassDescription` object might be replaced by a different version, such as when you have loaded a new provider of `NSClassDescription` objects, or when you are about to remove a provider of `NSClassDescription` objects.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

`NSClassDescription.h`

**registerClassDescription:forClass:**

Registers an `NSClassDescription` object for a given class in the `NSClassDescription` cache.

```
+ (void)registerClassDescription:(NSClassDescription *)description
    forClass:(Class)aClass
```

**Parameters**

*description*

The class description to register.

*aClass*

The class for which to register *description*.

**Discussion**

You should rarely need to directly invoke this method.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSClassDescription.h

## Instance Methods

### attributeKeys

Overridden by subclasses to return the names of attributes of instances of the described class.

- (NSArray \*)attributeKeys

**Return Value**

An array of NSString objects containing the names of attributes of instances of the described class.

**Discussion**

For example, a class description that describes Movie objects could return the attribute keys `title`, `dateReleased`, and `rating`.

If you have an instance of the class the receiver describes, you can use the NSObject instance method `attributeKeys` instead.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [toManyRelationshipKeys](#) (page 9)
- [toOneRelationshipKeys](#) (page 9)

**Declared In**

NSClassDescription.h

### inverseForRelationshipKey:

Overridden by subclasses to return the name of the inverse relationship from a relationship specified by a given key.

- (NSString \*)inverseForRelationshipKey:(NSString \*)*relationshipKey*

**Return Value**

The name of the inverse relationship from the relationship specified by *relationshipKey*.

**Discussion**

For a given key that defines the name of the relationship from the receiver's class to another class, returns the name of the relationship from the other class to the receiver's class. For example, suppose an Employee class has a relationship named `department` to a Department class, and that Department has a relationship named `employees` to Employee. The statement:

```
[employee inverseForRelationshipKey:@"department"];
```

returns the string `employees`.



If you have an instance of the class the receiver describes, you can use the `NSObject` instance method `inverseForRelationshipKey:` instead.

#### Availability

Available in Mac OS X v10.0 and later.

#### Declared In

`NSClassDescription.h`

## toManyRelationshipKeys

Overridden by subclasses to return the keys for the to-many relationship properties of instances of the described class.

- (NSArray \*)toManyRelationshipKeys

#### Return Value

An array of `NSString` objects containing the names of the to-many relationship properties of instances of the described class.

#### Discussion

To-many relationship properties are arrays of objects.

If you have an instance of the class the receiver describes, you can use the `NSObject` instance method `toManyRelationshipKeys` instead.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

- [attributeKeys](#) (page 8)
- [toOneRelationshipKeys](#) (page 9)

#### Declared In

`NSClassDescription.h`

## toOneRelationshipKeys

Overridden by subclasses to return the keys for the to-one relationship properties of instances of the described class.

- (NSArray \*)toOneRelationshipKeys

#### Return Value

An array of `NSString` objects containing the names of the to-one relationship properties of instances of the described class.

#### Discussion

To-one relationship properties are single objects.

If you have an instance of the class the receiver describes, you can use the `NSObject` instance method `toOneRelationshipKeys` instead.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [attributeKeys](#) (page 8)
- [toManyRelationshipKeys](#) (page 9)

**Declared In**

NSClassDescription.h

## Notifications

### NSClassDescriptionNeededForClassNotification

Posted by [classDescriptionForClass:](#) (page 6) when a class description cannot be found for a class.

After the notification is processed, [classDescriptionForClass:](#) (page 6) checks for a class description again. This checking allows an observer to register class descriptions lazily. The notification is posted only once for any given class, even if the class description remains undefined.

The notification object is the class object for which the class description is requested. This notification does not contain a *userInfo* dictionary.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSClassDescription.h

# Document Revision History

---

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

Date	Notes
2006-05-23	First publication of this content as a separate document.

**REVISION HISTORY**

Document Revision History

# Index

---

## A

---

attributeKeys **instance method** [8](#)

## C

---

classDescriptionForClass: **class method** [6](#)

## I

---

invalidateClassDescriptionCache **class method** [7](#)

inverseForRelationshipKey: **instance method** [8](#)

## N

---

NSClassDescriptionNeededForClassNotification  
**notification** [10](#)

## R

---

registerClassDescription:forClass: **class method**  
[7](#)

## T

---

toManyRelationshipKeys **instance method** [9](#)

toOneRelationshipKeys **instance method** [9](#)