

---

# ABGroup Class Reference

Data Management: Contact Data



2009-11-17



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

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

---

## **ABGroup Class Reference 5**

---

Overview	5
Tasks	5
Managing Properties	5
Managing Persons	6
Managing Subgroups	6
Managing Distribution Lists	6
Searching	6
Class Methods	6
addPropertiesAndTypes: properties	6 7
removeProperties:	7
searchElementForProperty:label:key:value:comparison: typeOfProperty:	8 8
Instance Methods	9
addMember:	9
addSubgroup:	10
distributionIdentifierForProperty:person:	10
members	11
parentGroups	11
removeMember:	11
removeSubgroup:	12
setDistributionIdentifier:forProperty:person:	13
subgroups	13

---

## **Document Revision History 15**

---

## **Index 17**

---



# ABGroup Class Reference

---

<b>Inherits from</b>	ABRecord : NSObject
<b>Conforms to</b>	NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/AddressBook.framework
<b>Availability</b>	Available in Mac OS X v10.2 and later.
<b>Companion guide</b>	Address Book Programming Guide for Mac OS X
<b>Declared in</b>	ABGroup.h

## Overview

The `ABGroup` class supports the concept of a “group” containing one or more persons. People may belong to multiple groups, and groups may also belong to other groups unless the relationship causes a circular reference. The only predefined property of a group is its name. However, similar to person records, you can add your own properties to group records. Groups not only help to organize person records, but also allow you to create email distribution lists.

The `ABGroup` class is “toll-free bridged” with its procedural C opaque-type counterpart. This means that the `ABGroupRef` type is interchangeable in function or method calls with instances of the `ABGroup` class.

## Tasks

### Managing Properties

- + [addPropertiesAndTypes:](#) (page 6)  
Adds the given properties to all records of this type in the Address Book database.
- + [removeProperties:](#) (page 7)  
Removes the given properties from all the records of this type in the Address Book database.
- + [properties](#) (page 7)  
Returns an array of the names of all the properties for this record type in the Address Book database.
- + [typeOfProperty:](#) (page 8)  
Returns the type for a given property.

## Managing Persons

- [addMember:](#) (page 9)  
Adds a person to a group.
- [removeMember:](#) (page 11)  
Removes a person from a group.
- [members](#) (page 11)  
Returns an array of persons in a group.

## Managing Subgroups

- [addSubgroup:](#) (page 10)  
Adds a subgroup to another group.
- [removeSubgroup:](#) (page 12)  
Removes a subgroup from a group.
- [parentGroups](#) (page 11)  
Returns an array containing a group's parents—that is, the groups that a group belongs to.
- [subgroups](#) (page 13)  
Returns an array containing a group's subgroups.

## Managing Distribution Lists

- [distributionIdentifierForProperty:person:](#) (page 10)  
Returns the distribution identifier for the given property and person.
- [setDistributionIdentifier:forProperty:person:](#) (page 13)  
Assigns a specific distribution identifier for a person's multivalue list property so that the group can be used as a distribution list.

## Searching

- + [searchElementForProperty:label:key:value:comparison:](#) (page 8)  
Returns a search element object that searches for records of this type.

## Class Methods

### **addPropertiesAndTypes:**

Adds the given properties to all records of this type in the Address Book database.

```
+ (NSInteger)addPropertiesAndTypes:(NSDictionary *)properties
```

**Parameters***properties*

A dictionary of properties to be added and their types.

**Return Value**

The number of properties successfully added.

**Discussion**

In each dictionary entry, the key is a string with the property's name, and the value is a constant with the property's type. The property's name must be unique. You may want to use Java-style package names for your properties, for example, `org.dogclub.dogname` or `com.mycompany.groupID`. The property type must be one of the constants described in "Property Types" in *Address Book Objective-C Constants Reference*.

**Availability**

Available in Mac OS X v10.2 and later.

**See Also**+ [removeProperties:](#) (page 7)**Declared In**

ABGroup.h

**properties**

Returns an array of the names of all the properties for this record type in the Address Book database.

+ (NSArray \*)*properties***Availability**

Available in Mac OS X v10.2 and later.

**See Also**+ [typeOfProperty:](#) (page 8)**Declared In**

ABGroup.h

**removeProperties:**

Removes the given properties from all the records of this type in the Address Book database.

+ (NSInteger)*removeProperties:(NSArray \*)properties***Parameters***properties*

An array of the properties to be removed.

**Return Value**

The number of properties successfully removed.

**Availability**

Available in Mac OS X v10.2 and later.

**See Also**

+ [addPropertiesAndTypes:](#) (page 6)

**Declared In**

ABGroup.h

**searchElementForProperty:label:key:value:comparison:**

Returns a search element object that searches for records of this type.

```
+ (ABSearchElement *)searchElementForProperty:(NSString *)property label:(NSString *)label key:(NSString *)key value:(id)value comparison:(ABSearchComparison)comparison
```

**Parameters**

*property*

The name of the property to search on. It cannot be `nil`. For a full list of the properties, see “Default Record Properties” in *Address Book Objective-C Constants Reference* and “Default Group Properties” in *Address Book Objective-C Constants Reference*.

*label*

The label name for a multivalue list. If *property* does not have multiple values, pass `nil`. If *property* does have multiple values, pass `nil` to search all the values. By default, ABGroup records don’t contain any multivalue list properties.

*key*

The key name for a dictionary. Pass `nil` if *property* is not a dictionary. If *property* is a dictionary, pass `nil` to search all keys. By default, ABGroup records don’t contain any properties that are dictionaries.

*value*

What you’re searching for. If `nil`, the only supported value for *comparison* is `kABEqual` or `kABNotEqual`.

*comparison*

The type of comparison to perform and is an `ABSearchComparison`, such as `kABEqual` or `kABPrefixMatchCaseInsensitive`.

**Availability**

Available in Mac OS X v10.2 and later.

**See Also**

+ `searchElementForProperty:label:key:value:comparison:` (`ABPerson`)  
 + `searchElementForConjunction:children:` (`ABSearchElement`)  
 - `recordsMatchingSearchElement:` (`ABAddressBook`)

**Declared In**

ABGroup.h

**typeOfProperty:**

Returns the type for a given property.

```
+ (ABPropertyType)typeOfProperty:(NSString *)property
```

### Parameters

*property*

The property whose type will be returned.

### Return Value

The property type of *property*.

### Discussion

If the property does not exist, this method returns `kABErrorInProperty`.

### Availability

Available in Mac OS X v10.2 and later.

### See Also

+ [properties](#) (page 7)

### Declared In

`ABGroup.h`

## Instance Methods

### **addMember:**

Adds a person to a group.

```
- (BOOL)addMember:(ABPerson *)person
```

### Parameters

*person*

The person record to be added.

### Return Value

YES if successful; otherwise, NO.

### Discussion

If the *person* argument is already part of the group, this method does nothing and returns NO. If the *person* argument is `nil`, this method raises an exception.

### Availability

Available in Mac OS X v10.2 and later.

### See Also

- [removeMember:](#) (page 11)

- [addSubgroup:](#) (page 10)

- [members](#) (page 11)

### Declared In

`ABGroup.h`

## addSubgroup:

Adds a subgroup to another group.

```
- (BOOL)addSubgroup:(ABGroup *)group
```

### Parameters

*group*

The group to add as a subgroup.

### Return Value

YES if successful; otherwise, NO.

### Discussion

If the *group* argument is already part of the receiver, this method does nothing and returns NO. If adding the group would create a recursion, this method also does nothing and returns NO. (For example, if the group Animal Lovers is in Dog Lovers, and you add Dog Lovers to Animal Lovers, that would create a recursion, which this method won't allow.) If the *group* argument is *nil*, this method raises an exception.

### Availability

Available in Mac OS X v10.2 and later.

### See Also

- [removeSubgroup:](#) (page 12)
- [addMember:](#) (page 9)
- [subgroups](#) (page 13)

### Declared In

ABGroup.h

## distributionIdentifierForProperty:person:

Returns the distribution identifier for the given property and person.

```
- (NSString *)distributionIdentifierForProperty:(NSString *)property person:(ABPerson *)person
```

### Parameters

*property*

The property whose distribution identifier will be returned.

*person*

The person whose distribution identifier will be returned.

### Return Value

The distribution identifier for the given property and person.

### Discussion

If a distribution identifier is not set, this method returns the multivalue's primary identifier. If either the *property* or the *person* argument is *nil*, this method returns *nil*. This method also returns *nil* if *property* is not a multivalue list property.

Distribution identifiers let you use groups as distribution lists, by indicating which value in a multivalue property should be used when addressing the group. See "Using Address Book Groups as Distribution Lists" for more detailed discussion.

### Availability

Available in Mac OS X v10.2 and later.

### See Also

- [setDistributionIdentifier:forProperty:person:](#) (page 13)
- [primaryIdentifier](#) (ABMultiValue)

### Declared In

ABGroup.h

## members

Returns an array of persons in a group.

- (NSArray \*)members

### Discussion

If this group doesn't contain any people, this method returns an empty array.

### Availability

Available in Mac OS X v10.2 and later.

### See Also

- [addMember:](#) (page 9)
- [removeMember:](#) (page 11)
- [subgroups](#) (page 13)

### Declared In

ABGroup.h

## parentGroups

Returns an array containing a group's parents—that is, the groups that a group belongs to.

- (NSArray \*)parentGroups

### Discussion

If this group doesn't belong to any groups, this method returns an empty array.

### Availability

Available in Mac OS X v10.2 and later.

### See Also

- [subgroups](#) (page 13)

### Declared In

ABGroup.h

## removeMember:

Removes a person from a group.

- (BOOL)removeMember:(ABPerson \*)*person*

#### Parameters

*person*

The person to be removed.

#### Return Value

YES if successful; otherwise, NO.

#### Discussion

If the *person* argument is not in the group, this method does nothing and returns NO. If *person* is *nil*, this method raises an exception.

#### Availability

Available in Mac OS X v10.2 and later.

#### See Also

- [addMember:](#) (page 9)
- [members](#) (page 11)
- [removeSubgroup:](#) (page 12)

#### Declared In

ABGroup.h

## removeSubgroup:

Removes a subgroup from a group.

- (BOOL)removeSubgroup:(ABGroup \*)*group*

#### Parameters

*group*

The subgroup to be removed.

#### Return Value

YES if successful; otherwise, NO.

#### Discussion

If the *group* argument is not a subgroup, this method does nothing and returns NO. If *group* is *nil*, this method raises an exception.

#### Availability

Available in Mac OS X v10.2 and later.

#### See Also

- [addSubgroup:](#) (page 10)
- [subgroups](#) (page 13)
- [removeMember:](#) (page 11)

#### Declared In

ABGroup.h

## setDistributionIdentifier:forProperty:person:

Assigns a specific distribution identifier for a person's multivalue list property so that the group can be used as a distribution list.

```
- (BOOL)setDistributionIdentifier:(NSString *)identifier forProperty:(NSString *)property person:(ABPerson *)person
```

### Parameters

*identifier*

The identifier to be set as the distribution identifier

*property*

The property whose distribution identifier will be set.

*person*

The person whose distribution identifier will be.

### Return Value

YES if successful; otherwise, NO.

### Discussion

The default distribution identifier is a multivalue list's primary identifier. If *person* is *nil*, this method raises an exception.

Distribution identifiers let you use groups as distribution lists, by indicating which value in a multivalue property should be used when addressing the group. See "Using Address Book Groups as Distribution Lists" for more detailed discussion.

### Availability

Available in Mac OS X v10.2 and later.

### See Also

- [distributionIdentifierForProperty:person:](#) (page 10)
- [setPrimaryIdentifier:](#) (ABMutableMultiValue)

### Declared In

ABGroup.h

## subgroups

Returns an array containing a group's subgroups.

```
- (NSArray *)subgroups
```

### Discussion

If this group doesn't contain any groups, this method returns an empty array.

### Availability

Available in Mac OS X v10.2 and later.

### See Also

- [addSubgroup:](#) (page 10)
- [removeSubgroup:](#) (page 12)
- [members](#) (page 11)

**Declared In**  
ABGroup.h

# Document Revision History

---

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

Date	Notes
2009-11-17	Fixed a broken link.
2009-07-23	Made minor editorial corrections throughout.
2009-06-02	Added descriptions of parameters and return values to API symbols, as needed.
2007-07-07	Fixed typos throughout.
2006-05-23	First publication of this content as a separate document.

**REVISION HISTORY**

Document Revision History

# Index

---

## A

---

addMember: **instance method** [9](#)  
addPropertiesAndTypes: **class method** [6](#)  
addSubgroup: **instance method** [10](#)

## D

---

distributionIdentifierForProperty:person:  
**instance method** [10](#)

## M

---

members **instance method** [11](#)

## P

---

parentGroups **instance method** [11](#)  
properties **class method** [7](#)

## R

---

removeMember: **instance method** [11](#)  
removeProperties: **class method** [7](#)  
removeSubgroup: **instance method** [12](#)

## S

---

searchElementForProperty:label:key:value:  
comparison: **class method** [8](#)  
setDistributionIdentifier:forProperty:person:  
**instance method** [13](#)  
subgroups **instance method** [13](#)

## T

---

typeOfProperty: **class method** [8](#)