
ABGroup Class Objective-C Reference

[Cocoa](#) > [Apple Applications](#)



2007-07-07



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, Cocoa, 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 Objective-C Reference 5

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

Document Revision History 15

Index 17

ABGroup Class Objective-C Reference

Inherits from	ABRecord : NSObject
Conforms to	NSObject (NSObject)
Framework	/System/Library/Frameworks/AddressBook.framework
Availability	Available in Mac OS X v10.2 and later.
Companion guide	Address Book Programming Guide for Mac OS X
Declared in	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 as long as the relationship does not cause 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 allow you to create email distribution lists.

Use the [members](#) (page 10) method to get all the members of a group, use the [addMember:](#) (page 9) method to add people to a group, and the [removeMember:](#) (page 11) method to remove people from a group. Use the [addSubgroup:](#) (page 9) method to create a subgroup.

Use the [addPropertiesAndTypes:](#) (page 7) method to add additional program-defined properties to group records. Because the Address Book database is stored as a property list, these program-defined properties may be ignored by other applications. Note that the Address Book database is accessed by multiple application and is not encrypted so your application should not store any sensitive information in the database.

You can also search for records matching a particular “query” you specify by creating an ABSearchElement object. Use the [searchElementForProperty:label:key:value:comparison:](#) (page 8) method to create an ABSearchElement object containing your query. Then use the `ABAddressBook recordsMatchingSearchElement:` method, passing the ABSearchElement as the argument, to query the database. See ABSearchElement for methods that create compound queries.

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 7)
- + [removeProperties:](#) (page 7)
- + [properties](#) (page 7)
- + [typeOfProperty:](#) (page 8)

Managing Persons

- [addMember:](#) (page 9)
- [removeMember:](#) (page 11)
- [members](#) (page 10)

Managing Subgroups

- [addSubgroup:](#) (page 9)
- [removeSubgroup:](#) (page 11)
- [parentGroups](#) (page 10)
- [subgroups](#) (page 12)

Distribution Lists

- [distributionIdentifierForProperty:person:](#) (page 10)
- [setDistributionIdentifier:forProperty:person:](#) (page 12)

Searching

+ [searchElementForProperty:label:key:value:comparison:](#) (page 8)

Class Methods

addPropertiesAndTypes:

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

Discussion

Adds the given properties to all records of this type in the Address Book database, and returns the number of properties successfully added. 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 "Constants" in ABRecord.

Availability

Available in Mac OS X v10.2 and later.

See Also

+ [removeProperties:](#) (page 7)

Declared In

ABGroup.h

properties

+ (NSArray *)*properties*

Discussion

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

Availability

Available in Mac OS X v10.2 and later.

See Also

+ [typeOfProperty:](#) (page 8)

Declared In

ABGroup.h

removeProperties:

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

Discussion

Removes the given properties from all the records of this type in the Address Book database, and returns the number of properties successfully removed.

Availability

Available in Mac OS X v10.2 and later.

See Also

+ [addPropertiesAndTypes:](#) (page 7)

Declared In

ABGroup.h

searchElementForProperty:label:key:value:comparison:

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

Discussion

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

- *property* is the name of the property to search on. It cannot be `nil`. For a full list of the properties, see “Constants” (page 12) and “Constants” in `ABRecord`.
- *label* is the label name for a multi-value 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 multi-value list properties.
- *key* is the key name for a dictionary. If *property* is not a dictionary, pass `nil`. If *property* is a dictionary, pass `nil` to search all keys. By default, `ABGroup` records don’t contain any properties that are dictionaries.
- *value* is what you’re searching for. If `nil`, then the only supported value for *comparison* is `kABEqual` or `kABNotEqual`.
- *comparison* specifies 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:

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


Discussion

Returns the type for a given property. 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:

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

Discussion

Adds a person to a group. Returns YES if successful. If the *person* argument is already part of 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

- [removeMember:](#) (page 11)
- [addSubgroup:](#) (page 9)
- [members](#) (page 10)

Declared In

ABGroup.h

addSubgroup:

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

Discussion

Adds a subgroup to another group. Returns YES if successful. 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 11)

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

Declared In
ABGroup.h

distributionIdentifierForProperty:person:

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

Discussion

Returns the distribution identifier for the given property and person. If a distribution identifier is not set, this method returns the multi-value's primary identifier. If either the *property* or *person* arguments are *nil*, this method returns *nil*. Also, returns *nil* if *property* is not a multi-value list property.

Availability

Available in Mac OS X v10.2 and later.

See Also

- [setDistributionIdentifier:forProperty:person:](#) (page 12)
- [primaryIdentifier](#) (ABMultiValue)

Declared In
ABGroup.h

members

- (NSArray *)members

Discussion

Returns an array of persons in a group. 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 12)

Declared In
ABGroup.h

parentGroups

- (NSArray *)parentGroups

Discussion

Returns an array containing a group's parents—the groups that a group belongs to. 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 12)

Declared In

ABGroup.h

removeMember:

– (BOOL)removeMember:(ABPerson *)*person*

Discussion

Removes a person from a group. Returns YES if successful. 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 10)

– [removeSubgroup:](#) (page 11)

Declared In

ABGroup.h

removeSubgroup:

– (BOOL)removeSubgroup:(ABGroup *)*group*

Discussion

Removes a subgroup from a group. Returns YES if successful. 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 9)

– [subgroups](#) (page 12)

– [removeMember:](#) (page 11)

Declared In

ABGroup.h

setDistributionIdentifier:forProperty:person:

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

Discussion

Assigning a specific distribution identifier for a person’s multi-value list property so that the group can be used as a distribution list (mailing list, in the case of an email property). The default distribution identifier is a multi-value list’s primary identifier. Use this method if you need to change the distribution identifier for a particular person. For example, if the default identifier is a person’s home email but you want to use John’s work email, invoke this method passing `kABEmailWorkLabel` as the *identifier* argument, `kABEmailProperty` as the *property* argument, and John’s person object as the *person* argument. Pass `nil` for the *identifier* argument to reset the distribution identifier to its default. If *person* is `nil`, this method raises an exception. Returns `YES` if successful, `NO` otherwise.

Availability

Available in Mac OS X v10.2 and later.

See Also

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

Declared In

ABGroup.h

subgroups

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

Discussion

Returns an array containing a group’s subgroups. 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 9)
- [removeSubgroup:](#) (page 11)
- [members](#) (page 10)

Declared In

ABGroup.h

Constants

This is one of the properties that an ABGroup record contains by default. ABRecord describes some properties that all records contain, in “Constants”

Constant	Description
kABGroupNameProperty	Name of the group

Document Revision History

This table describes the changes to *ABGroup Class Objective-C Reference*.

Date	Notes
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 7](#)
addSubgroup: [instance method 9](#)

D

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

K

kABGroupNameProperty [constant 13](#)

M

members [instance method 10](#)

P

parentGroups [instance method 10](#)
properties [class method 7](#)

R

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

S

searchElementForProperty:label:key:value:
 comparison: [class method 8](#)
setDistributionIdentifier:forProperty:person:
 [instance method 12](#)
subgroups [instance method 12](#)

T

typeOfProperty: [class method 8](#)