

---

# ABRecord Class Reference

Data Management: Contact Data



2009-07-23



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, 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

---

## **ABRecord Class Reference 5**

---

Overview 5

Tasks 5

    Initializing Records 5

    Retrieving and Setting Values 6

    Retrieving a Specific Record 6

    Determining Properties 6

Instance Methods 6

    init 6

    initWithAddressBook: 6

    isReadOnly 7

    removeValueForProperty: 7

    setValue:forProperty: 7

    uniqueId 8

    valueForProperty: 9

---

## **Document Revision History 11**

---

## **Index 13**

---



# ABRecord Class Reference

---

<b>Inherits from</b>	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>	ABRecord.h
<b>Related sample code</b>	AddressBookCocoa CocoaPeoplePicker

## Overview

`ABRecord` is an abstract superclass providing a common interface to, and defining common properties for, all Address Book records. A property is a field in the database record, such as the first or last name of a person record. `ABRecord` defines the types of properties supported, and basic methods for getting, setting, and removing property values.

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

## Tasks

### Initializing Records

- [initWithAddressBook:](#) (page 6)  
Initializes a record using the given address book.
- [init](#) (page 6)  
Initializes a record using the shared address book.

## Retrieving and Setting Values

- [removeValueForProperty:](#) (page 7)  
Removes the value for a given property.
- [setValue:forProperty:](#) (page 7)  
Sets the value of a given property for a record.
- [valueForProperty:](#) (page 9)  
Returns the value of a given property for a record.

## Retrieving a Specific Record

- [isReadOnly](#) (page 7)  
Returns whether a record is read-only.

## Determining Properties

- [uniqueId](#) (page 8)  
Returns the unique ID for a record.

# Instance Methods

### **init**

Initializes a record using the shared address book.

```
- (id) init
```

#### **Discussion**

On Mac OS X 10.5 and later, this is equivalent to `initWithAddressBook:[ABAddressBook addressBook]`.

#### **Availability**

Available in Mac OS X v10.5 and later.

#### **Declared In**

ABRecord.h

### **initWithAddressBook:**

Initializes a record using the given address book.

```
- (id) initWithAddressBook:(ABAddressBook *) addressBook
```

#### **Parameters**

*addressBook*

The address book with which to initialize the record.

**Discussion**

The record is added to *addressBook* but is not visible to other address books until *addressBook* is saved. This method is the designated initializer for this class.

**Availability**

Available in Mac OS X v10.5 and later.

**Declared In**

ABRecord.h

**isReadOnly**

Returns whether a record is read-only.

```
- (BOOL) isReadOnly
```

**Return Value**

YES if the record is read-only; otherwise, NO.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

ABRecord.h

**removeValueForProperty:**

Removes the value for a given property.

```
- (BOOL) removeValueForProperty:(NSString *) property
```

**Parameters**

*property*

The property whose value will be removed.

**Return Value**

YES if the value is removed successfully; otherwise, NO.

**Discussion**

When you next call [valueForProperty:](#) (page 9) on that property, it returns *nil*.

If *property* is *nil*, this method raises an exception.

**Availability**

Available in Mac OS X v10.2 and later.

**Declared In**

ABRecord.h

**setValue:forProperty:**

Sets the value of a given property for a record.

```
- (BOOL) setValue:( id) value forProperty:(NSString *) property
```

**Parameters***value*

The value to set for property.

*property*

The property whose value will be set.

**Return Value**

YES if the value was set successfully; otherwise, NO.

**Discussion**

The type of the value must match the property's type (see "Property Types" in *Address Book Objective-C Constants Reference* for a list of possible property types). If *property* is `nil` or if *value* is not of the correct type, this method raises an exception. If *property* is a multivalued list property, this method checks to see if the values in the multivalued list are the same type. If the multivalued list contains mixed types, the value will not be set successfully.

**Availability**

Available in Mac OS X v10.2 and later.

**Related Sample Code**

AddressBookCocoa

**Declared In**

ABRecord.h

## uniqueId

Returns the unique ID for a record.

```
- (NSString *) uniqueId
```

**Return Value**

The unique ID.

**Discussion**

This method is equivalent to invoking [valueForProperty:](#) (page 9), passing `kABUIDProperty` as the argument.

**Availability**

Available in Mac OS X v10.2 and later.

**See Also**

```
- recordForUniqueId:(ABAddressBook)
```

**Related Sample Code**

Birthdays

**Declared In**

ABRecord.h

## valueForProperty:

Returns the value of a given property for a record.

```
- (id) valueForProperty:(NSString *) property
```

### Parameters

*property*

The property whose value will be returned.

### Return Value

The value of the given property.

### Discussion

The type of the value depends on the property type (see “Property Types” in *Address Book Objective-C Constants Reference* for a list of possible property types). Note that the returned value is always of an immutable type (for example, an `NSString` type, not an `NSMutableString` type, is returned).

If *property* is `nil`, this method raises an exception.

### Availability

Available in Mac OS X v10.2 and later.

### Related Sample Code

AddressBookCocoa

Birthdays

### Declared In

ABRecord.h



# Document Revision History

---

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

Date	Notes
2009-07-23	Added descriptions of parameters and return values to API symbols, as needed.
2009-06-02	Added descriptions of parameters and return values to API symbols, as needed.
2006-07-26	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

---

## I

---

init [instance method 6](#)  
initWithAddressBook: [instance method 6](#)  
isReadOnly [instance method 7](#)

## R

---

removeValueForProperty: [instance method 7](#)

## S

---

setValue:forProperty: [instance method 7](#)

## U

---

uniqueId [instance method 8](#)

## V

---

valueForProperty: [instance method 9](#)