
SKPaymentTransactionObserver Protocol Reference

Networking & Internet



2010-01-20



Apple Inc.
© 2010 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, and iPhone are trademarks of Apple Inc., registered in the United States and other countries.

iOS is a trademark or registered trademark of Cisco in the U.S. and other countries and is used under license.

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

SKPaymentTransactionObserver Protocol Reference 5

Overview 5

Tasks 5

 Handling Transactions 5

 Handling Restored Transactions 5

Instance Methods 6

 paymentQueue:removedTransactions: 6

 paymentQueue:restoreCompletedTransactionsFailedWithError: 6

 paymentQueue:updatedTransactions: 6

 paymentQueueRestoreCompletedTransactionsFinished: 7

Document Revision History 9

SKPaymentTransactionObserver Protocol Reference

Conforms to	NSObject
Framework	/System/Library/Frameworks/StoreKit.framework
Availability	Available in iOS 3.0 and later.
Companion guide	In App Purchase Programming Guide
Declared in	SKPaymentQueue.h

Overview

The `SKPaymentTransactionObserver` protocol declares methods that are implemented by observers of an `SKPaymentQueue` object.

An observer is called when transactions are updated by the queue or removed from the queue. An observer should process all successful transactions, unlock the functionality purchased by the user, and then finish the transaction by calling the payment queue's `finishTransaction:` method.

Tasks

Handling Transactions

- [paymentQueue:updatedTransactions:](#) (page 6) *required method*
Tells an observer that one or more transactions have been updated. (required)
- [paymentQueue:removedTransactions:](#) (page 6)
Tells an observer that one or more transactions have been removed from the queue.

Handling Restored Transactions

- [paymentQueue:restoreCompletedTransactionsFailedWithError:](#) (page 6)
Tells the observer that an error occurred while restoring transactions.
- [paymentQueueRestoreCompletedTransactionsFinished:](#) (page 7)
Tells the observer that the payment queue has finished sending restored transactions.

Instance Methods

paymentQueue:removedTransactions:

Tells an observer that one or more transactions have been removed from the queue.

```
- (void)paymentQueue:(SKPaymentQueue *)queue removedTransactions:(NSArray *)transactions
```

Parameters

queue

The payment queue that updated the transactions.

transactions

An array of the transactions that were removed.

Discussion

Your application does not typically need to implement this method but might implement it to update its own user interface to reflect that a transaction has been completed.

Availability

Available in iOS 3.0 and later.

Declared In

SKPaymentQueue.h

paymentQueue:restoreCompletedTransactionsFailedWithError:

Tells the observer that an error occurred while restoring transactions.

```
- (void)paymentQueue:(SKPaymentQueue *)queue restoreCompletedTransactionsFailedWithError:(NSError *)error
```

Parameters

queue

The payment queue that was restoring transactions.

error

The error that occurred.

Availability

Available in iOS 3.0 and later.

Declared In

SKPaymentQueue.h

paymentQueue:updatedTransactions:

Tells an observer that one or more transactions have been updated. (required)

```
- (void)paymentQueue:(SKPaymentQueue *)queue updatedTransactions:(NSArray *)transactions
```

Parameters*queue*

The payment queue that updated the transactions.

transactions

An array of the transactions that were updated.

Discussion

The application should process each transaction by examining the transaction's `transactionState` property. If `transactionState` is `SKPaymentTransactionStatePurchased`, payment was successfully received for the desired functionality. The application should make the functionality available to the user. If `transactionState` is `SKPaymentTransactionStateFailed`, the application can read the transaction's error property to return a meaningful error to the user.

Once a transaction is processed, it should be removed from the payment queue by calling the payment queue's `finishTransaction:` method, passing the transaction as a parameter.

Important: Once the transaction is finished, Store Kit can not tell you that this item is already purchased. It is important that applications process the transaction completely before calling `finishTransaction:`.

Availability

Available in iOS 3.0 and later.

Declared In

SKPaymentQueue.h

paymentQueueRestoreCompletedTransactionsFinished:

Tells the observer that the payment queue has finished sending restored transactions.

```
- (void)paymentQueueRestoreCompletedTransactionsFinished:(SKPaymentQueue *)queue
```

Parameters*queue*

The payment queue that restored the transactions.

Discussion

This method is called after all restorable transactions have been processed by the payment queue. Your application is not required to do anything in this method.

Availability

Available in iOS 3.0 and later.

Declared In

SKPaymentQueue.h

Document Revision History

This table describes the changes to *SKPaymentTransactionObserver Protocol Reference*.

Date	Notes
2010-01-20	Fixed a typo.
2009-11-17	Marked which methods in the protocol are optional.
2009-07-13	Added descriptions of restored transaction methods.
2009-05-01	Minor edits.
2009-03-12	New document that describes an interface for observing transactions updated by the payment queue.

REVISION HISTORY

Document Revision History