Using Application Loader

2011-08-17

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

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

Introduction

This guide describes Application Loader 2.4.1, an Apple tool to help you prepare your applications for sale in the App Store. You can use Application Loader 2.4.1 to create an Application Loader package containing the In-App Purchase metadata you want to add to your application and to submit your application binary files to the App Store.

Application Loader offers fast uploading, stable connections, and early validation warnings. Make sure the status of your application in iTunes Connect is “Waiting for Upload” before you use Application Loader to submit your binary files to the App Store.

You can also use Application Loader to create and send a package containing a new iOS hardware accessories prototype application you want to test.

What’s New

Application Loader 2.4.1 contains the following enhancements:

- **Improved user interface.** Application Loader 2.4.1 includes a new “Welcome to Application Loader” window, “Choose an Activity” window, Hardware Accessory Prototype Application window for prototypes, and *SKU*.itmsp Application Loader windows for In-App Purchases. Application Loader 2.4.1 also includes updated Deliver Your App windows, as well as improved error messaging.
- **In-App Purchase delivery.** Using Application Loader you can now create an Application Loader package containing the In-App Purchase metadata you want to add to your application. You can also use Application Loader to create and deliver multiple In-App Purchases at once.
- **Hardware accessories prototype testing.** You can use Application Loader to create and send a package containing a new iOS hardware accessories prototype application you want to test.
- **Updated system requirements.** In addition to newly supported formats, Application Loader 2.4.1 contains updated server, IP addresses, and port information.
- **Send to Apple command**. If you encounter an error while delivering a package to Apple, Application Loader creates a log report for you to send to Apple to help with troubleshooting your delivery issues.

Organization of This Document

This guide is organized as follows:

- "Getting Started" (page 7), provides an overview of Application Loader, including system requirements, supported formats, and the steps you must take to install and start using Application Loader.
- "Creating Your In-App Purchase" (page 13), describes how to create an Application Loader package containing an In-App Purchase to submit to iTunes.
- "Describing Your In-App Purchase" (page 21), explains how to prepare your Application Loader package containing an In-App Purchase for delivery to iTunes by providing In-App Purchase metadata.
- "Delivering Your In-App Purchase" (page 29), provides details on how to deliver your Application Loader package containing an In-App Purchase to iTunes.
- "Delivering Your App Binary" (page 31), provides details on how to deliver your application binary files to the App Store.
- "Creating, Describing, and Delivering Your Prototype" (page 33), describes how to create and send a package containing a new iOS hardware accessories prototype application you want to test.
- "Common Questions" (page 37), features some of the common questions you might have while using Application Loader.
- "Glossary" (page 43), defines terms you'll encounter as you read this guide.

In addition to the sections listed above, a revision history and index are provided for your reference.

Getting Additional Information

If you have any questions regarding Application Loader, review the Frequently Asked Questions in iTunes Connect at http://www.apple.com/itunes/go/itunesconnect/faqs.

If you need to request technical assistance or would like to provide feedback about Application Loader, visit the Contact Us page in iTunes Connect at http://www.apple.com/itunes/go/itunesconnect/contactus.

Click ? in the "Choose an Activity" window to display this *Using Application Loader* guide.

AppleCare support is not available for this product.

Getting Started

This section provides an overview of Application Loader, including system requirements, supported formats, and the steps you must take to install and start using Application Loader.

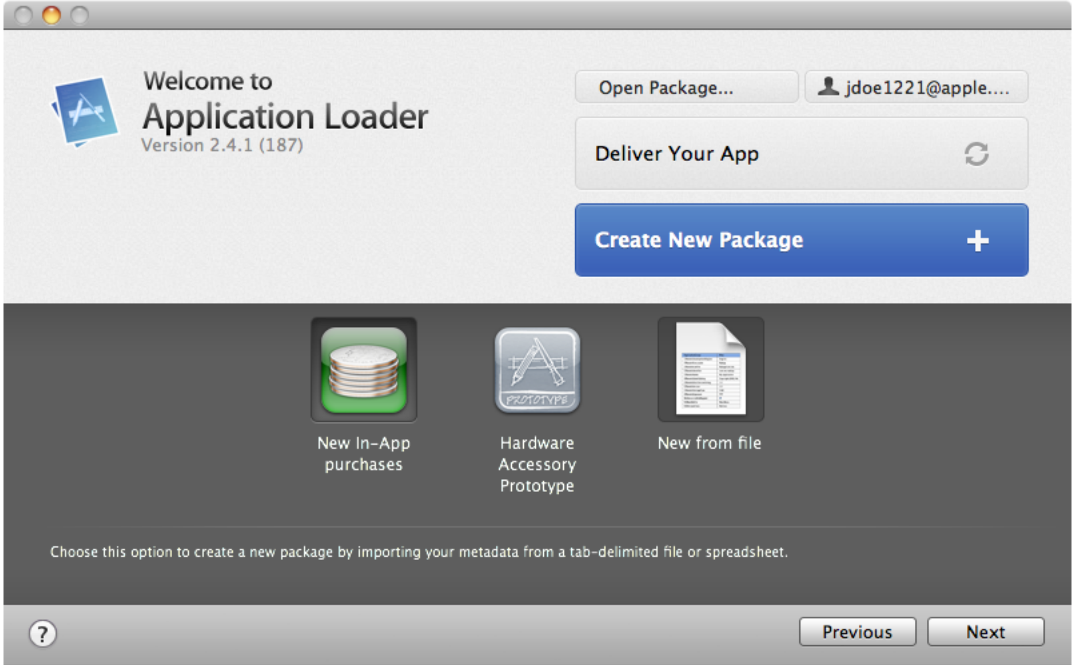
What is Application Loader?

Application Loader is a tool to help you prepare your applications for sale in the App Store. You can use Application Loader 2.4.1 to create an Application Loader package containing the In-App Purchase metadata you want to add to your application and to submit your application binary files to the App Store.

Application Loader offers fast uploading, stable connections, and early validation warnings. Make sure the status of your application in iTunes Connect is "Waiting for Upload" before you use Application Loader to submit your binary files to the App Store.

You can also use Application Loader to create and send a package containing a new iOS hardware accessories prototype application you want to test.

The following screen shot displays the Application Loader "Choose an Activity" window.



Choose this option to create a new package by importing your metadata from a tab-delimited file or spreadsheet.

System and Network Requirements

Application Loader 2.4.1 requires an Intel-based Macintosh with at least 512 MB RAM and Mac OS X version 10.6.8 or later installed.

Apple recommends at least 20 GB of available hard disk space and a broadband internet connection with an upload rate of 1MB/sec or faster.

Note: Because Application Loader is integrated with the Xcode developer tools package you can also deliver binary files directly from Xcode. For more information on this delivery mechanism, review the Distributing Applications section in the *iOS Development Guide* located in the Apple Dev Center.

The following table lists details about the servers Application Loader uses to deliver packages and application binary files. For the best upload experience, verify that all of the ports and IP addresses are accessible.

Server	IP Address	TCP Port	UDP Port
itmsdav.apple.com	17.152.19.125	443	n/a
contentdelivery.itunes.apple.com	17.152.19.127	443*	n/a
vgr101.apple.com	17.152.249.51	33001	33001-33500
vgr102.apple.com	17.152.249.52	33001	33001-33500
vgr103.apple.com	17.152.249.53	33001	33001-33500
vgr104.apple.com	17.152.249.54	33001	33001-33500
vgr105.apple.com	17.152.249.55	33001	33001-33500
vgr106.apple.com	17.152.249.56	33001	33001-33500
vgr107.apple.com	17.152.249.57	33001	33001-33500
vgr108.apple.com	17.152.249.58	33001	33001-33500
vgr301.apple.com	17.172.190.141	33001	33001-33500
vgr302.apple.com	17.172.190.142	33001	33001-33500
vgr303.apple.com	17.172.190.143	33001	33001-33500
vgr304.apple.com	17.172.190.144	33001	33001-33500
vgr305.apple.com	17.172.190.145	33001	33001-33500
vgr306.apple.com	17.172.190.146	33001	33001-33500
vgr307.apple.com	17.172.190.147	33001	33001-33500
vgr308.apple.com	17.172.190.148	33001	33001-33500
sgr201.apple.com	17.152.249.101	44001	44001-44500

Server	IP Address	TCP Port	UDP Port
sgr202.apple.com	17.152.249.102	44001	44001-44500
sgr203.apple.com	17.152.249.103	44001	44001-44500
sgr204.apple.com	17.152.249.104	44001	44001-44500
sgr205.apple.com	17.152.249.105	44001	44001-44500
sgr206.apple.com	17.152.249.106	44001	44001-44500
sgr207.apple.com	17.152.249.107	44001	44001-44500
sgr208.apple.com	17.152.249.108	44001	44001-44500
sgr401.apple.com	17.172.190.101	44001	44001-44500
sgr402.apple.com	17.172.190.102	44001	44001-44500
sgr403.apple.com	17.172.190.103	44001	44001-44500
sgr404.apple.com	17.172.190.104	44001	44001-44500
sgr405.apple.com	17.172.190.105	44001	44001-44500
sgr406.apple.com	17.172.190.106	44001	44001-44500
sgr407.apple.com	17.172.190.107	44001	44001-44500
sgr408.apple.com	17.172.190.108	44001	44001-44500

Notes:

- * Indicates a required port. To deliver content, you must open at least one of the ports.
- Because Apple periodically releases new versions and updates to its software, servers listed in the table might change mid-release.

Supported Formats

Application Loader supports ZIP files produced by Finder (passing a code sign check upon upload) for delivering your application binary and PKG package files for delivering your In-App Purchase. For specific details, refer to the *Developer Reference Library* in the Apple Dev Center for creating properly signed ZIP files.

Screenshots should be a clear picture of your In-App Purchase in action and be a 320 x 460, 320 x 480, 480 x 320, or 480 x 300 pixel TIFF, JPEG, or PNG file in RGB Color mode.

You can use your keyboard to perform many of the same actions you perform using the mouse. With full keyboard access, you use the Tab key, arrow keys, and Space bar to move to and select or enable items on the screen.

Installing Application Loader

When you install the Xcode developer tools package, the installation package automatically installs Application Manager 2.4.1 files in your /Developer/Applications/Utilities folder.

Open the Application Loader application by double-clicking the "Application Loader.app" icon in the /Developer/Applications/Utilities folder. You are now ready to begin using Application Loader. For more information, see "Starting Application Loader" (page 10) later in this section.

Note: The Xcode developer tools package is part of the latest Software Development Kit (SDK) from Apple. You can download the latest SDK when you sign in to the Apple Dev Center.

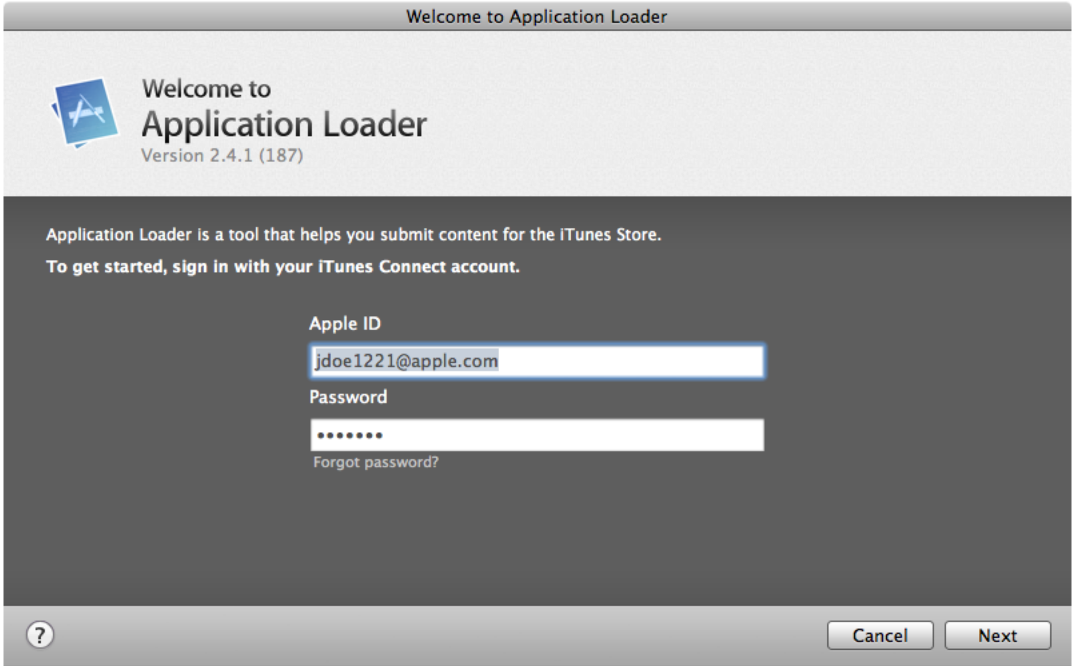
Starting Application Loader

Before you can use Application Loader to create an Application Loader package and to submit your application binary files to the App Store, you must initialize Application Loader by providing your iTunes Connect login information at startup.

1. Double-click the "Application Loader.app" icon in the /Developer/Applications/Utilities folder folder.

2. Read the Software License Agreement, and then click Accept.

   Application Loader opens the "Welcome to Application Loader" window.



Welcome to Application Loader

Welcome to
Application Loader
Version 2.4.1 (187)

Application Loader is a tool that helps you submit content for the iTunes Store.

To get started, sign in with your iTunes Connect account.

Apple ID: jdoe1221@apple.com

Password: •••••••

3. Type your iTunes Connect login information (Apple ID and password) in the "Welcome to Application Loader" window, and then click Next.

   Application Loader verifies your iTunes Connect login information and stores your Apple ID and password for future Application Loader operations.

Initializing Application Loader with your iTunes Connect login information is account-specific. If you share your machine with other users with different iTunes Connect login information, you can each have your own Apple ID and password setting for Application Loader.

4. Click Next.

Application Loader displays the "Choose an Activity" window so you can begin using Application Loader. For more information, see "Delivering Your App Binary" (page 31) later in this guide.

If the "Choose an Activity" window does not display any activities, you do not have an active contract with iTunes. You must complete the contract process before you can deliver packages to iTunes and submit application binary files to the App Store. For more information, see the Contracts, Tax and Banking module in iTunes Connect at http://www.apple.com/itunes/go/itunesconnect/contracts.

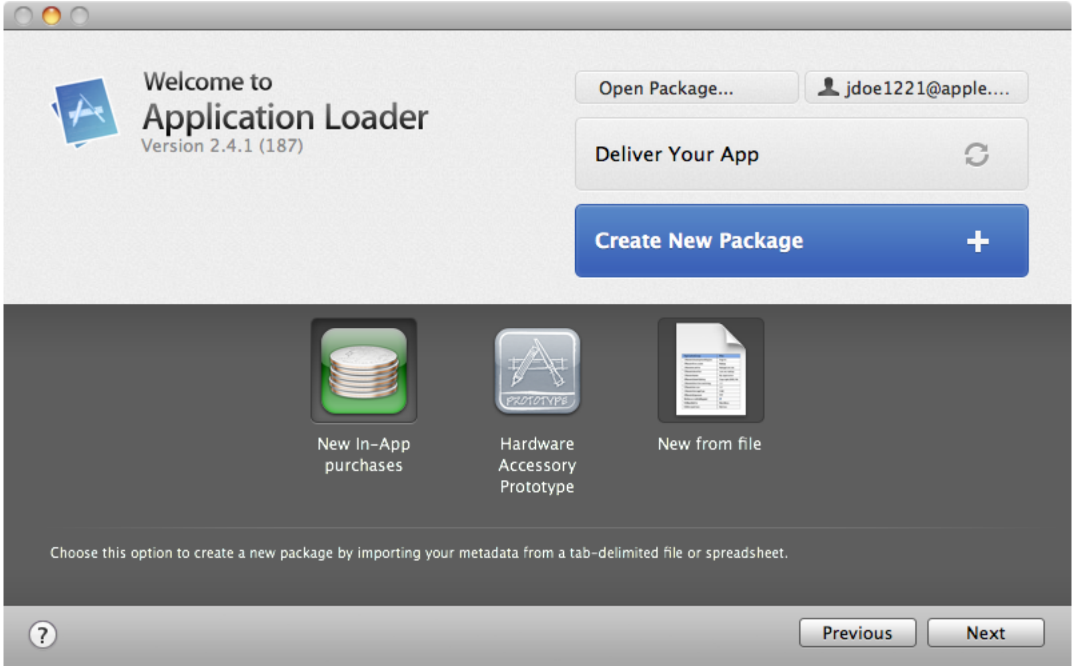
Notes:

- If you change your iTunes Connect password, you must use the "Welcome to Application Loader" window to reset your Apple ID and password in Application Loader. To reset your Apple ID and password, choose Window > Sign In As in Application Loader, and then repeat steps 2 through 4 above.
- The main Application Loader windows contain an active user button displaying the Apple ID for the currently logged in user. Click the button to display the "Welcome to Application Loader" window to change or update your Apple ID and password.
- Click ? to display this *Using Application Loader* guide.

Creating Your In-App Purchase

The first step in adding In-App Purchase metadata to your application is to create a package containing the metadata you want to submit. You can create the package containing the metadata by creating a new In-App Purchase, from an existing Application Loader package, or by importing metadata details from a file.

The following screen shot displays the Application Loader "Choose an Activity" window with the "New from file" option selected.



Choose this option to create a new package by importing your metadata from a tab-delimited file or spreadsheet.

If you do not see any of the above activities in the Application Loader "Choose an Activity" window, you might not have configured Application Loader for delivery. Click the active user button in the "Choose an Activity" window to display the "Welcome to Application Loader" window again.

Important: Apple enables the "New In-App purchases" option on a developer-by-developer basis. You cannot use this option without prior authorization from Apple. For more information, contact the App Store Team using the Contact Us page in iTunes Connect at http://www.apple.com/itunes/go/itunesconnect/contactus.

Creating a New In-App Purchase

You can create a package containing the In-App Purchase you want to submit to iTunes by creating a new In-App Purchase.

To add metadata using a new In-App Purchase:

1. Click Create New Package in the "Choose an Activity" window.

   The "Choose an Activity" window appears automatically when you open Application Loader, or you can choose File > New to display the "Choose an Activity" window from within Application Loader.

2. Select "New In-App purchases", and then click Next.

   Application Loader displays the Manage In-App Purchases window displaying a list of all your existing applications.

   Click Refresh to update the list of applications with the most recent information from iTunes Connect.

   Use the search field to filter In-App Purchases in the table by App Name, Apple ID, SKU, App Type, and In-App Purchases information.

3. Select the application where you want to add In-App Purchases, and then click Manage.

   Application Loader displays a *SKU*.itmsp Application Loader window with the In-App Purchases pane selected.

4. Use the In-App Purchases pane to add new In-App Purchases with metadata information, rights and pricing, and languages. For more information, see "Describing Your In-App Purchase" (page 21).

When you are sure the In-App Purchase details are complete and correct, you can send the package to iTunes. For more information, see "Delivering Your In-App Purchase" (page 29).

Important: Apple enables the "New In-App purchases" option on a developer-by-developer basis. You cannot use this option without prior authorization from Apple. For more information, contact the App Store Team using the Contact Us page in iTunes Connect at http://www.apple.com/itunes/go/itunesconnect/contactus.

Opening an Existing Package

You can create a package containing the In-App Purchase you want to submit to iTunes by opening a previously saved Application Loader package and adding and editing metadata.

To add metadata from an existing package:

1. Click Open Package in the "Choose an Activity" window.

   Application Loader displays an Open dialog so you can select an existing package.

   The "Choose an Activity" window appears automatically when you open Application Loader, or you can choose File > New to display the "Choose an Activity" window from within Application Loader.

2. Select the package name containing the metadata you want to include in your application, and then click Open.

   Application Loader opens the package and displays a *SKU*.itmsp Application Loader window with the In-App Purchases pane selected.

3. Use the In-App Purchases pane to add new In-App Purchases and add and edit metadata information, rights and pricing, and languages. For more information, see "Describing Your In-App Purchase" (page 21).

   You can also replace assets you want to update (for example, screenshots).

Note: If you previously delivered the package to iTunes, you cannot update all information in the In-App Purchases pane. For more information, see "What elements can I update in my In-App Purchase?" (page 41)

When you are sure the In-App Purchase details are complete and correct, you can send the package to iTunes. For more information, see "Delivering Your In-App Purchase" (page 29).

Creating a Package from a File

You can create a package containing the In-App Purchase you want to submit to iTunes by importing details from an Application Loader file and adding and editing the metadata.

When creating a package from a file, be sure to use the Application Loader template file provided to you. For more information, contact the App Store Team using the Contact Us page in iTunes Connect at http://www.apple.com/itunes/go/itunesconnect/contactus.

To import metadata details from a file:

1. Click Create New Package in the "Choose an Activity" window.

   The "Choose an Activity" window appears automatically when you open Application Loader, or you can choose File > New to display the "Choose an Activity" window from within Application Loader.

2. Select "New from file", and then click Next.

   Application Loader displays an Open dialog so you can select the file containing the metadata details you want to import into the new package.

3. Select the filename containing the metadata details you want to include in the new package, and then click Open.

   Note: Any fields Application Loader cannot import are left blank, including misspelled field names or fields not acceptable for import.

4. Click Yes to import multiple screenshots at once, if applicable.

   When you create a package using "New from file", you can import multiple In-App Purchases at once and therefore import multiple screenshots for these In-App Purchases at once. You can choose the images individually or select an entire directory. Screenshot filenames must match the In-App Purchase product identifiers (for example, *product_identifier*.png). For example, if a product identifier is com.cyberinteractive.touchfighter.missiles.fifty the matching screenshot filename must be com.cyberinteractive.touchfighter.missiles.fifty.png.

   Application Loader imports the details from the Application Loader file and displays a *SKU*.itmsp Application Loader window with the In-App Purchases pane selected. If there is more than one SKU in the file, Application Loader imports the details and displays a *SKU*.itmsp Application Loader window, with the In-App Purchases pane selected, for each unique SKU.

5. Use the In-App Purchases pane to add new In-App Purchases and add and edit metadata information, rights and pricing, and languages. For more information, see "Describing Your In-App Purchase" (page 21).

When you are sure the In-App Purchase details are complete and correct, you can send the package to iTunes. For more information, see "Delivering Your In-App Purchase" (page 29).

You can import the following fields into the package (required fields in **bold**):

Name	Description	Example
SKU	The SKU for the application associated with the In-App Purchase. The SKU must match an existing application SKU for the currently logged in user.	com.cyberinteractive.touchfighter.3
Reference Name	A unique name (from 1 to 255 characters) for the In-App Purchase. The reference name must be unique within the application. The reference name is displayed in iTunes Connect and in sales and trends reports.	30 missiles
Product ID	A unique identifier (from 2 to 100 characters) for the In-App Purchase. The product identifier must be unique across all your applications and add-ons and contain only alphanumeric characters, underscores, and periods. Application Loader uses the product identifier in reporting.	com.cyberinteractive.touchfighter.missiles.thirty
Type	The type of In-App Purchase you want to add. You can add consumable, non-consumable, and non-renewing subscription In-App Purchases. Application Loader does not currently support auto-renewable subscription In-App Purchases.	Consumable
Cleared For Sale	Indicates whether or not the In-App Purchase should be made available for sale.	yes
Wholesale Price Tier	A retail price tier (from 1 to 85) for the current In-App Purchase. Multiple price tier intervals are not supported when you create a package from a file. You can enter one interval with one wholesale price tier, one effective date, and one end date.	3

Name	Description	Example
Displayed Name @@language@@	A user-visible display name (from 2 to 75 characters) where @@language@@ is a supported localized language. You can add a new column for each localized language display name.	30 missiles (English) 30 missiles (French)
Description @@language@@	A user-visible description (from 10 to 255 characters) where @@language@@ is a supported localized language. You can add a new column for each localized language description.	Buy thirty extra missiles to help you on your way. (English) Acheter trente missiles supplémentaires pour vous aider sur votre chemin. (French)
Screenshot Path	The local path and filename for the associated screenshot for review. Screenshots must be a 320 x 460, 320 x 480, 480 x 320, or 480 x 300 pixel TIFF, JPEG, or PNG file in RGB Color mode.	/Documents/ApplicationLoader/wwdc/screenshots/com.cyberinteractive.touchfighter.missiles.thirty.jpg
Effective Date	The date when the new price tier takes effect for the current In-App Purchase. Application Loader changes the tier at the beginning of the day. To immediately change the tier, specify Now. You must specify a future date in the current or following year for price effective date.	Jun 6, 2011
End Date	The date when the new price tier reverts to the previously specified price tier for the current In-App Purchase. Application Loader changes the tier at the beginning of the day. For example, a one day price reduction does not begin and end on the same day. Instead, it would end at the start of the following day. To indefinitely change the tier, specify None. You must specify a future date in the current or following year for price end date. The price end date you specify must be after the price effective date.	Jun 10, 2011

Notes:

- You can import multiple applications at once and there is no limit to the number of In-App Purchases you can include in an Application Loader template file. However, Application Loader supports a maximum of 3000 In-App Purchases per application.
- You can exclude any column, as long as you leave the column heading intact.
- Only include fields for the data you want to import. Application Loader imports empty or blank fields as blank data and overwrites any existing data.
- All rows of data must be contiguous. To import data successfully, there can be no blank rows of data between any two rows of data.
- If you are using Excel to save your data as a Unicode Text UTF-16 file, be sure to review the data before you save the file. Excel can automatically remove all leading zeros from your SKU numbers and also save the numbers in scientific notation. Reviewing your data before you save the file as a Unicode Text UTF-16 file can help avoid this issue. To ensure your SKU numbers do not have leading zeros removed and are not displayed in scientific notation, first select the SKU column in your spreadsheet. Choose Format > Cells and select Text for the category. Now you can properly enter your SKU numbers in the cells.

Describing Your In-App Purchase

After you create the package for the In-App Purchase you want to add to your application, but before the package is ready to be sent as an Application Loader package to iTunes, you must prepare the In-App Purchase by describing the metadata details you want to submit, including information, rights and pricing, and languages.

Providing In-App Purchase Details

Before sending a package to iTunes, use the In-App Purchases pane of Application Loader to provide In-App Purchase metadata information, rights and pricing, and language details.

Entering In-App Purchase Information

Use the In-App Purchases > Info pane to provide as much detail as possible about your In-App Purchase.

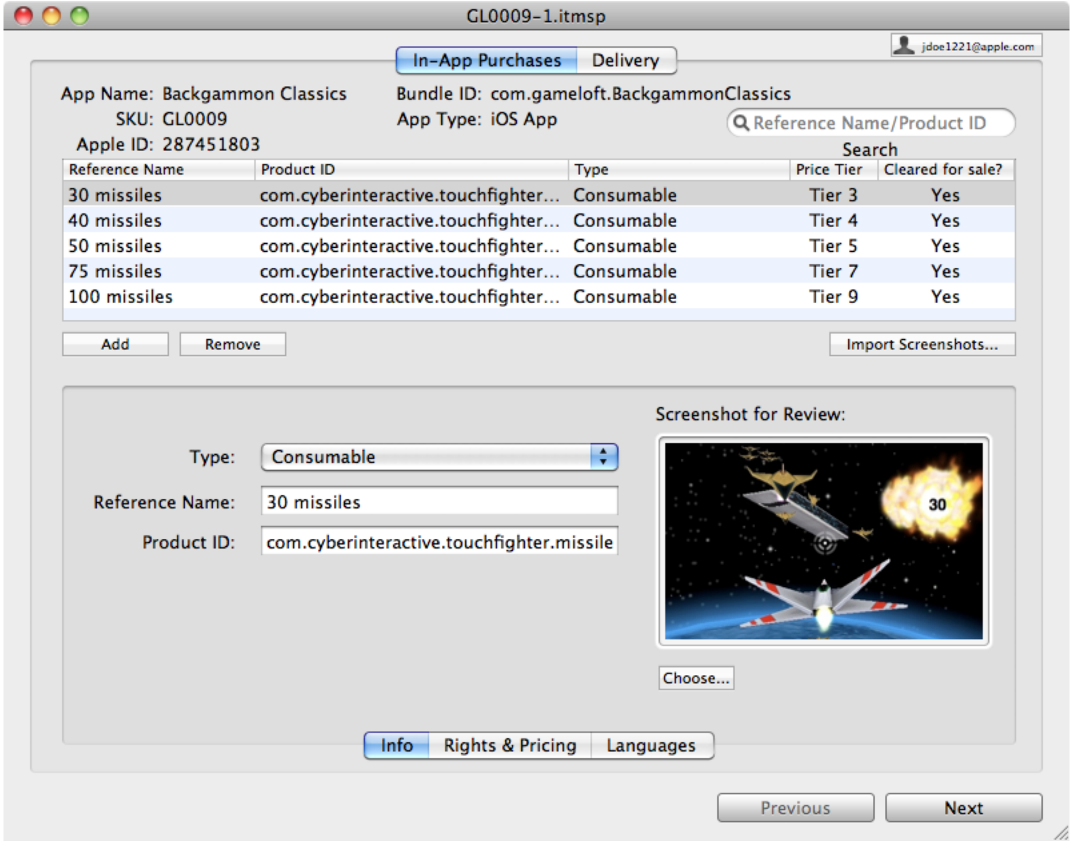
To provide In-App Purchase information details:

1. Select an existing In-App Purchase or click Add to add a new In-App Purchase for the current application.

   Click Remove to delete the selected In-App Purchase. Application Loader removes the In-App Purchase from the table but does not delete it from iTunes Connect. Application Loader supports a maximum of 3000 In-App Purchases per application.

   Use the search field to filter In-App Purchases in the table by Reference Name, Product ID, Type, and Price Tier information.

2. Click Info at the bottom of the In-App Purchases pane.



App Name: Backgammon Classics
SKU: GL0009
Apple ID: 287451803
Bundle ID: com.gameloft.BackgammonClassics
App Type: iOS App

Reference Name	Product ID	Type	Price Tier	Cleared for sale?
30 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 3	Yes
40 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 4	Yes
50 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 5	Yes
75 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 7	Yes
100 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 9	Yes

Type: Consumable
Reference Name: 30 missiles
Product ID: com.cyberinteractive.touchfighter.missile

3. Choose the type of In-App Purchase you want to add from the Type pop-up menu:

   - **Consumable** (available for iOS and Mac OS X applications). Users must purchase consumable products each time they need the item. For example, one-time services are commonly implemented as consumable products.
   - **Non-Consumable** (available for iOS and Mac OS X applications). Users purchase non-consumable products only once. Once a non-consumable product is purchased, it is provided to all devices associated with the user's iTunes account. Store Kit provides built-in support to restore non-consumable products on multiple devices.
   - **Non-Renewing Subscription** (available for iOS applications only). A non-renewing subscription shares attributes of consumable and non-consumable products. Like a consumable product, users can purchase a non-renewing subscription multiple times; allowing developers to implement their own renewal mechanism in the application. Subscriptions must be provided on all devices associated with a user. The Application Loader In-App Purchase feature expects subscriptions to be delivered through an external server provided by developers. Developers must provide the infrastructure to deliver subscriptions to multiple devices.

   Application Loader does not currently support auto-renewable subscription In-App Purchases.

   You cannot edit the In-App Purchase type after your In-App Purchase is created in the Store.

4. Enter details for the In-App Purchase:

   - **Reference Name:** Type a unique 1 to 255 character name for the In-App Purchase. The reference name must be unique within the application. The reference name is displayed in iTunes Connect and in sales and trends reports.
   - **Product ID:** Type a unique 2 to 100 character identifier for the In-App Purchase. The product identifier must be unique across all your applications and add-ons and contain only alphanumeric characters, underscores (_), and periods (.). For example, com.company.app_name.productid. Application Loader uses the product identifier in reporting. You cannot edit your product identifier after submitting your In-App Purchase. Once you use a product identifier for one application purchase, you cannot use it again.

5. Import one or more screenshots (review purposes only) for the In-App Purchase(s):

   - Drag a screenshot to the “Screenshot for Review” box, or click Choose and select a screenshot file for the In-App Purchase. To replace a previously added file, drag a new file to the "Screenshot for Review” box or click Choose to select a new file, or choose Edit > Delete to delete the file and then add a new file.
   - Click Import Screenshots to import multiple screenshots for multiple In-App Purchases. Each In-App Purchase can have only one screenshot associated with it. Application Loader uses the screenshots you import in your application product pages.

   Screenshots must be a 320 x 460, 320 x 480, 480 x 320, or 480 x 300 pixel TIFF, JPEG, or PNG file in RGB Color mode.

   The screenshot filename must match the currently selected In-App Purchase product identifier (for example, *product_identifier*.png). For example, if the product identifier is com.cyberinteractive.touchfighter.missiles.fifty the matching screenshot filename must be com.cyberinteractive.touchfighter.missiles.fifty.png.

   Do not scale up artwork if it is smaller than the required size. If you are scanning your screenshot be sure to scan at the highest resolution, or at least 320 x 460 pixels. Screenshots must be at least 72 dots per inch (dpi). Do not submit art that contains an embedded color profile, back cover, or thumbnail art.

6. Click Next or Rights & Pricing at the bottom of the In-App Purchases pane. For more information, see “Entering In-App Purchase Rights and Pricing” (page 24).

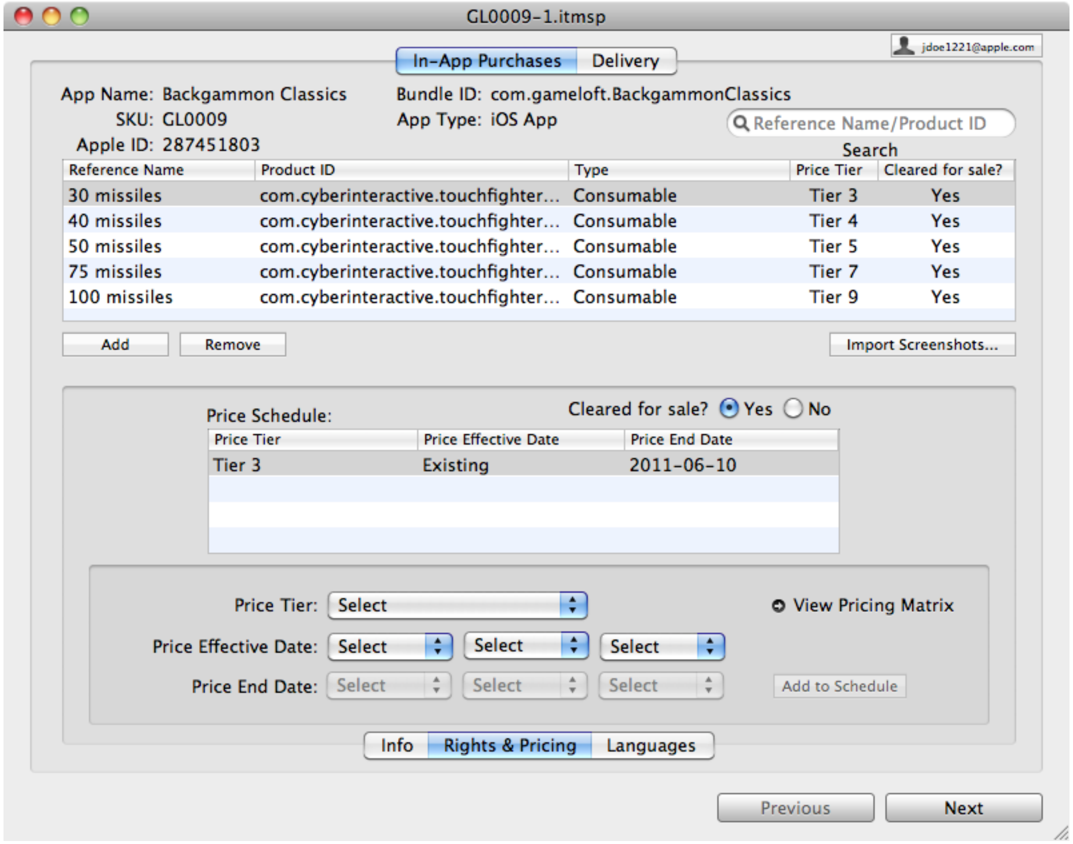
Entering In-App Purchase Rights and Pricing

Use the In-App Purchases > Rights & Pricing pane to provide as much detail as possible about the rights and pricing of your In-App Purchase.

You can set the price tier and schedule price tier changes for the future by setting an effective date and a end date. Application Loader provides a pricing matrix to assist with your pricing decisions.

To provide In-App Purchase rights and pricing details:

1. Click Rights & Pricing at the bottom of the In-App Purchases pane.



App Name: Backgammon Classics
SKU: GL0009
Apple ID: 287451803
Bundle ID: com.gameloft.BackgammonClassics
App Type: iOS App

Reference Name	Product ID	Type	Price Tier	Cleared for sale?
30 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 3	Yes
40 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 4	Yes
50 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 5	Yes
75 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 7	Yes
100 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 9	Yes

Price Schedule:
Cleared for sale? ● Yes ○ No

Price Tier	Price Effective Date	Price End Date
Tier 3	Existing	2011-06-10

Price Tier: Select
Price Effective Date: Select Select Select
Price End Date: Select Select Select

2. Select an existing In-App Purchase or click Add to add a new In-App Purchase in the current application.

   Click Remove to delete the selected In-App Purchase. Application Loader removes the In-App Purchase from the table but does not delete it from iTunes Connect. Application Loader supports a maximum of 3000 In-App Purchases per application.

   Use the search field to filter In-App Purchases in the table by Reference Name, Product ID, Type, and Price Tier information.

3. Use the “Cleared for sale” radio buttons to indicate whether or not the In-App Purchase should be made available for sale.

   Click No to remove the In-App Purchase from your application in the App Store.

4. Choose an initial retail price tier (from 1 to 85) for the current In-App Purchase from the Price Tier pop-up menu, and then click "Add to Schedule".

   Application Loader adds the price tier to the Price Schedule table and updates the table to reflect the new price intervals.

   Note: The first time your choose a retail price tier you cannot choose a price effective date or price end date. For the first retail price tier, Application Loader automatically sets the effective date to Existing and the end date to None.

5. Choose another retail price tier (from 1 to 85) for the current In-App Purchase from the Price Tier pop-up menu.

6. Choose the date when the new price tier takes effect for the current In-App Purchase from the Price Effective Date pop-up menus. Application Loader changes the tier at the beginning of the day. To immediately change the tier, choose Now from the pop-up menus. You must specify a future date in the current or following year for price effective date. For example, present 2011 and 2012 with the latest date of December 31, 2012.

7. Choose the date when the new price tier reverts to the previously specified price tier for the current In-App Purchase from the Price End Date pop-up menus. Application Loader changes the tier at the beginning of the day. For example, a one day price reduction does not begin and end on the same day. Instead, it would end at the start of the following day. To indefinitely change the tier, choose None from the pop-up menus. You must specify a future date in the current or following year for price end date. For example, present 2011 and 2012 with the latest date of December 31, 2012. The price end date you specify must be after the price effective date.

8. Click "Add to Schedule".

   Application Loader adds the price tier to the Price Schedule table and updates the table to reflect the new price intervals.

9. Repeat steps 5 through 8 for each price tier you want to add to the selected In-App Purchase.

   Click View Pricing Matrix to display all App Store pricing matrix information.

10. Repeat steps 2 through 8 for each price tier you want to add to each In-App Purchase in the current application.

11. Click Next or Languages at the bottom of the In-App Purchases pane. For more information, see "Entering In-App Purchase Languages" (page 26).

Notes:

- You can set up as many price tier changes as you want and your application will change prices on your effective dates.
- The Price Tier column displays the wholesale price tier. However, if you set up multiple price tier intervals, the Price Tier column for the In-App Purchase displays "multiple" to indicate that multiple price tiers are in use.

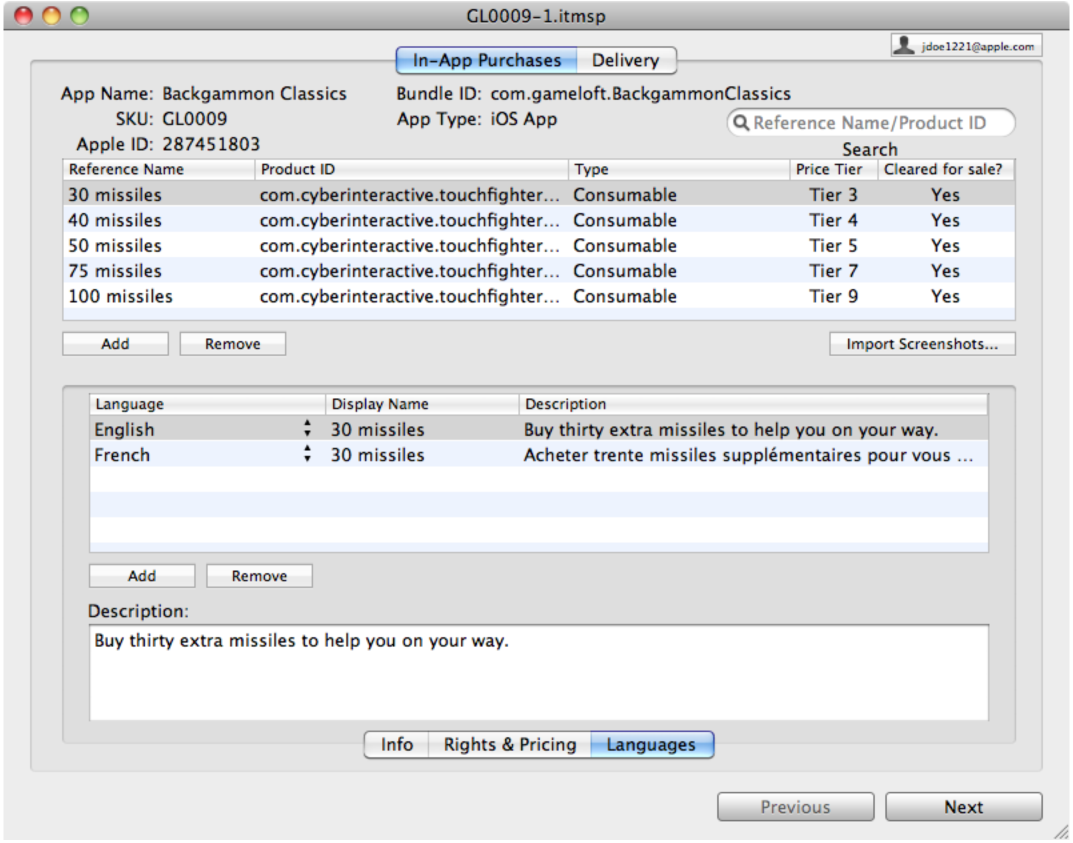
Entering In-App Purchase Languages

Use the In-App Purchases > Languages pane to provide as much detail as possible about the language, display name, and description text for your In-App Purchase.

You can specify additional languages in which you want to enter your application metadata in addition to the English (or other default language you set) metadata. For example, if you choose to upload new metadata in French, the text appears in all French-speaking stores you specified.

To provide In-App Purchase language details:

1. Click Languages at the bottom of the In-App Purchases pane.



GL0009-1.itmsp

App Name: Backgammon Classics
SKU: GL0009
Apple ID: 287451803
Bundle ID: com.gameloft.BackgammonClassics
App Type: iOS App

Reference Name	Product ID	Type	Price Tier	Cleared for sale?
30 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 3	Yes
40 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 4	Yes
50 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 5	Yes
75 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 7	Yes
100 missiles	com.cyberinteractive.touchfighter...	Consumable	Tier 9	Yes

Language	Display Name	Description
English	30 missiles	Buy thirty extra missiles to help you on your way.
French	30 missiles	Acheter trente missiles supplémentaires pour vous ...

Description:
Buy thirty extra missiles to help you on your way.

2. Select an existing In-App Purchase or click Add to add a new In-App Purchase for the current application.

Click Remove to delete the selected In-App Purchase. Application Loader removes the In-App Purchase from the table but does not delete it from iTunes Connect. Application Loader supports a maximum of 3000 In-App Purchases per application.

Use the search field to filter In-App Purchases in the table by Reference Name, Product ID, Type, and Price Tier information.

3. Click Add to add a language for the current In-App Purchase, and then choose a language from the pop-up menu. The language indicates both the language and the specific location where the language is spoken. You should provide a language and a localized display name and description for each country where the In-App Purchase is available.

   Click Remove to delete the selected language.

4. Type a localized display name in the Display Name column for the In-App Purchase, and then press Return. Display names can be 2 to 75 characters in length, can include Rich Text formatting, and must be suitable for public display.

5. Type a localized description in the Description column or box for the In-App Purchase, and then press Return. Descriptions can be 10 to 255 characters in length, can include Rich Text formatting, and must be suitable for public display.

6. Repeat steps 3 through 5 to add additional languages, display names, and descriptions for the selected In-App Purchase.

7. Repeat steps 2 through 6 for each language you want to add to each In-App Purchase in the current application.

8. Click Next at the bottom of the In-App Purchases pane. For more information, see "Delivering Your In-App Purchase" (page 29).

Delivering Your In-App Purchase

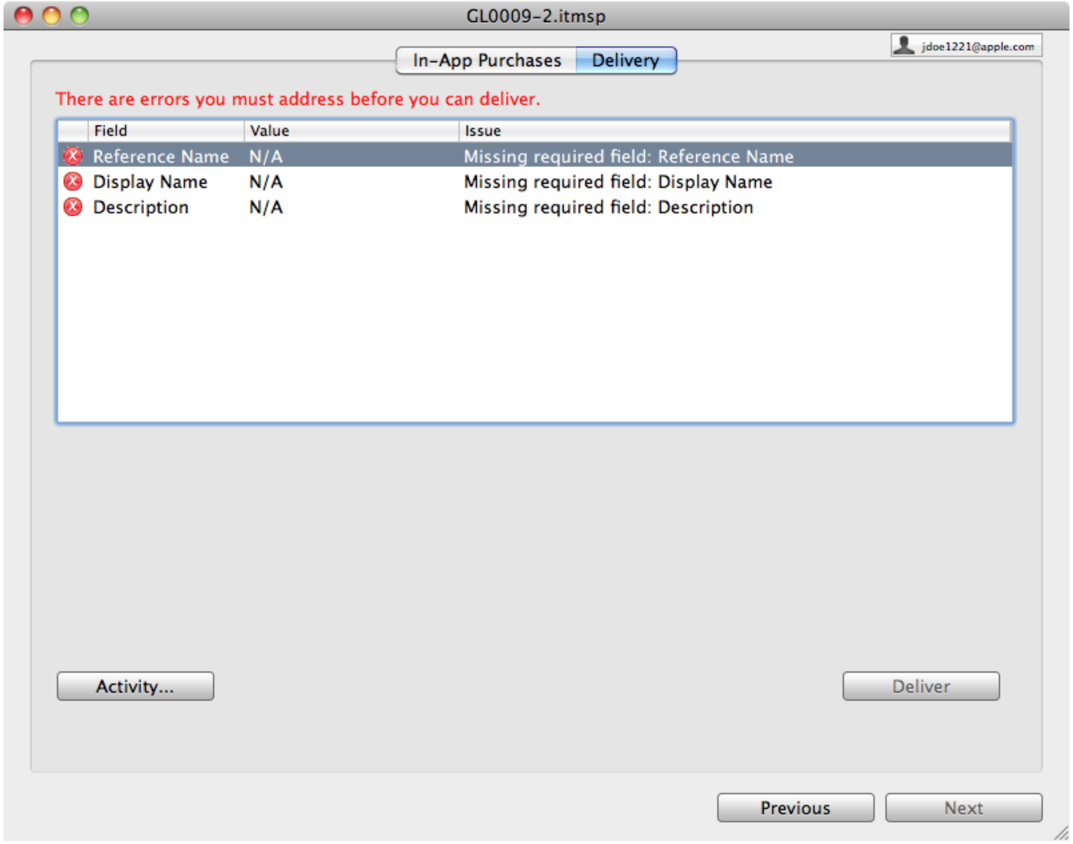
The final step in adding your In-App Purchase to your application is to deliver the Application Loader package to iTunes. Once you have created the package and entered all your In-App Purchase metadata details, you can deliver the package and all your prepared files as an Application Loader package directly to iTunes. This section describes how to send your Application Loader package to iTunes and how to send Apple a log report if you encounter errors during package delivery.

Sending a Package to iTunes

When you are sure a package is complete and correct, including all In-App Purchase metadata details, you can send the package and all your prepared files as an Application Loader package to iTunes.

To deliver a package:

1. Click the Delivery pane for the package in the Application Loader window.



There are errors you must address before you can deliver.

Field	Value	Issue
Reference Name	N/A	Missing required field: Reference Name
Display Name	N/A	Missing required field: Display Name
Description	N/A	Missing required field: Description

Application Loader processes all the In-App Purchase metadata details and displays any warnings or errors.

2. Fix any delivery warnings or errors.

   Double-click on a warning or error to have Application Loader jump to the exact location of the problem.

   Click Activity to display the Background Activity window and view delivery status information for all packages you are currently sending, including pending messages, time remaining, warning and error messages, and so on.

3. Click Deliver.

   Application Loader packages all your In-App Purchase metadata details and prepared files into an Application Loader package and sends it to iTunes.

   You can also choose File > "Save and Deliver" to save and package all your In-App Purchase metadata details and prepared files into an Application Loader package and send it to iTunes. Or, to send multiple packages at once, choose File > "Save and Deliver All". Application Loader saves and packages all your In-App Purchase metadata details and prepared files, for all open Application Loader windows, into Application Loader packages and sends them to iTunes.

Notes:

- Until Apple has imported all your metadata, you can continue to update In-App Purchase metadata details and resend packages.
- Application Loader plays an alert sound when the package delivery is complete. If Application Loader encounters an error during delivery, Application Loader plays an error sound and bounces the Application Loader icon in the Dock.
- To modify your delivery settings, choose Application Loader > Preferences, click Advanced, and then deselect Signiant, Aspera, or DAV. You should only change these settings with the advice of Apple. For any questions, contact the App Store Team using the Contact Us page in iTunes Connect at http://www.apple.com/itunes/go/itunesconnect/contactus.

Sending Your Log Report to Apple

If you encounter an error while delivering a package to Apple, Application Loader creates a log report for you to send to Apple to help with troubleshooting your delivery issues.

To send the log report:

- Click "Send to Apple" in the Delivery pane to deliver the error log to the Application Loader support team.

  To view the details of the error log before sending, click 🔍, and then click "Send to Apple".

Delivering Your App Binary

When you have finished uploading your assets and completed the application submission process in iTunes Connect, and your application is in the “Waiting for Upload” state, you can deliver your application binary files to the App Store.

This section describes how to submit your application binary files to the App Store.

Submitting Your Application Binary Files

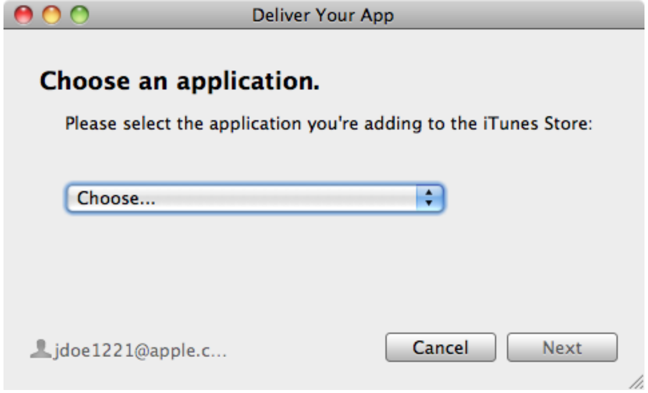
When the status of your application in iTunes Connect is “Waiting for Upload”, you can use Application Loader to submit your binary files to the App Store.

To submit your application binary files:

1. Click Deliver Your App in the “Choose an Activity” window.

   The “Choose an Activity” window appears automatically when you open Application Loader, or you can choose File > New to display the “Choose an Activity” window from within Application Loader.

2. Choose the name of your application from the Choose pop-up menu.



Deliver Your App

Choose an application.

Please select the application you're adding to the iTunes Store:

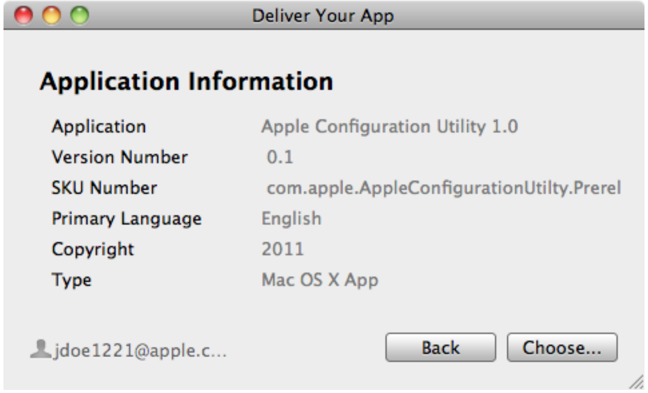
   Your application appears in the Choose pop-up menu when the application status in iTunes Connect is “Waiting for Upload”. If you have uploaded your metadata and screenshots into iTunes Connect but have not yet uploaded your binary, or if your application has been rejected, you must click "Ready to Upload Binary" in iTunes Connect before you will have eligible applications to upload. Clicking "Ready to Upload Binary" changes your application’s status to "Waiting for Upload".

3. Click Next.

   Application Loader searches for all eligible applications which are ready to be added to the App Store, and then displays a Please Confirm window.

4. Click Yes to indicate you have tested and qualified your application.

   Application Loader displays a window with your application details, including name, version number, SKU number, primary language, copyright, and type.



Application Information

Application	Apple Configuration Utility 1.0
Version Number	0.1
SKU Number	com.apple.AppleConfigurationUtilty.Prerel
Primary Language	English
Copyright	2011
Type	Mac OS X App

5. Click Choose, select your application file, and then click Open.

   Click the ellipsis button to replace the current file before submitting.

6. Click Send.

   Application Loader begins submitting your application binary file to the App Store.

   Click Cancel to stop the upload and close the window. Click ⊗ to stop the transfer and keep the window open to select another file to submit.

7. Click Done.

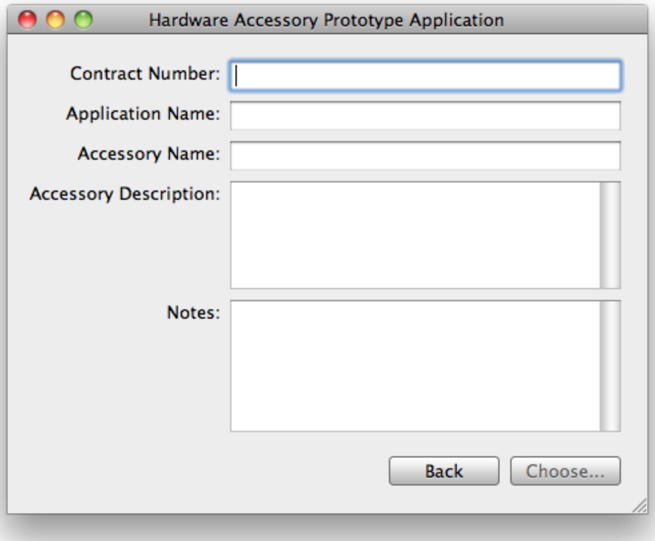
   Use the Manage Your Applications page in iTunes Connect to view the status of your application.

Choose Window > Background Activity to display the Background Activity window and view delivery status information for all applications, including pending messages, time remaining, warning and error messages, and so on.

Creating, Describing, and Delivering Your Prototype

Application Loader helps you prepare your applications for sale in the App Store. In addition to using Application Loader to create anApplication Loader package containing the In-App Purchase metadata you want to add to your application and to submit your application binary files to the App Store, you can also use Application Loader to create and send a package containing a new iOS hardware accessories prototype application you want to test.

The following screen shot displays the Application Loader Hardware Accessory Prototype Application window.



Creating a Prototype Package

If you have the correct contracts with iTunes, you can create a package containing a new iOS hardware accessories prototype application and send it to the App Store for testing.

To select a prototype file on your computer:

1. Click Create New Package in the “Choose an Activity” window.

   The “Choose an Activity” window appears automatically when you open Application Loader, or you can choose File > New to display the “Choose an Activity” window from within Application Loader.

2. Select Hardware Accessory Prototype, and then click Next.

   Application Loader displays the Hardware Accessory Prototype Application window.

3. Read the wizard information, and then click Next.

4. Use the Hardware Accessory Prototype Application window to specify your prototype details (for example, contract number, application name, accessory name and description, and notes). For more information, see "Describing a Prototype Package" (page 34).

When you are sure the prototype details are complete and correct, you can send the package to the App Store for testing. For more information, see "Delivering a Prototype Package" (page 35).

Describing a Prototype Package

After you create the package for the iOS hardware accessories prototype application you want to test, but before the package is ready to be sent as an Application Loader package to the App Store, you must prepare the prototype by describing the iOS hardware accessories prototype in detail, including contract number, application name, accessory name and description, and notes.

Providing Prototype Details

Before sending your iOS hardware accessories prototype application to the App Store, use the Hardware Accessory Prototype Application window to provide prototype details.

Entering Prototype Information

Use the Hardware Accessory Prototype Application window to provide as much detail as possible about the iOS hardware accessories prototype application.

To provide prototype information details:

1. Enter prototype information:

   Contract Number: Type a unique identifier for the prototype application. Apple recommends using the SKU of your application as the contract number. Contract numbers can be numeric, up to 100 ASCII characters, and do not need to be a fixed number of digits. Although contract numbers can contain digits, Application Loader treats them as strings, not numbers. Thus, a contract number of "00000000012345" is not the same as "12345". Contract numbers can only contain uppercase alphanumeric characters, underscores (_), and dashes (-), and cannot contain spaces, other punctuation, or symbols. Contract Number is case-insensitive.

   Application Name: Type a unique 1 to 255 character name for the prototype application you are adding to the App Store. For best results in application product pages, specify a name that is no more than 70 characters long (35 characters for the device App Store.

   Accessory Name: Type a unique 1 to 255 character name for the hardware accessory (for example, a blue tooth head set) you are testing with your prototype application.

   Accessory Description: Type a description for the hardware accessory (for example, a blue tooth head set) you are testing with your prototype application.

Notes: Type any notes and additional information you want to send about your prototype application. For example, you might want to list new features, user interface improvements, or bug fixes you specifically want to test.

2. Click Choose, select your prototype application file, and then click Open.

   Application Loader displays a submitting application window so you can verify your prototype application file before you send it to the App Store.

Delivering a Prototype Package

The final step before testing your iOS hardware accessories prototype application is to deliver the Application Loader package to the App Store. Once you have created the package and entered all your iOS hardware accessories prototype application details, you can deliver the package as an Application Loader package directly to Apple's App Store.

Sending Your Prototype Package to the App Store

When you are sure a package is complete and correct, including all iOS hardware accessories prototype application details, you can send the package as an Application Loader package to the App Store.

To deliver your prototype package:

1. Click Send in the submitting application window.

   Click the ellipsis button to replace the current file before submitting.

   Application Loader packages all your iOS hardware accessories prototype application details into an Application Loader package and sends it to the App Store.

2. Click Done.

   Use the Manage Your Applications page in iTunes Connect to view the status of your iOS hardware accessories prototype application.

Choose Window > Background Activity to display the Background Activity window and view delivery status information for all packages, including pending messages, time remaining, warning and error messages, and so on.

Common Questions

This section features some of the common questions you might have while using Application Loader.

What is Application Loader?

Application Loader is a Cocoa application that assists you with preparing your applications for sale in the App Store. You can use Application Loader 2.4.1 to create an Application Loader file containing the In-App Purchase metadata you want to add to your application and to submit your application binary files to the App Store.

Application Loader offers fast uploading, a more stable connection than the iTunes Connect interface, and early validation warnings. Make sure the status of your application in iTunes Connect is "Waiting for Upload" before you use Application Loader to submit your binary files to the App Store.

You can also use Application Loader to create and send a package containing a new iOS hardware accessories prototype application you want to test.

I don't think I have Application Loader currently installed. How can I get it?

When you install the Xcode developer tools package, the installation package automatically installs Application Manager 2.4.1 files in your /Developer/Applications/Utilities folder.

Open the Application Loader application by double-clicking the "Application Loader.app" icon in the /Developer/Applications/Utilities folder. You are now ready to begin using Application Loader. For more information, see "Starting Application Loader" (page 10).

After changing my password for my iTunes Connect login, I can no longer run Application Loader. I have re-installed my Application Loader and the problem remains the same. What should I do?

Application Loader uses the same login credentials as iTunes Connect. In order to change the login credentials you set for Application Loader, you must display the "Welcome to Application Loader" window. To access the "Welcome to Application Loader" window, choose Window > Sign In As in Application Loader.

Which operating system is required to run Application Loader?

Application Loader 2.4.1 requires an Intel-based Macintosh with at least 512 MB RAM and Mac OS X version 10.6.8 or later installed.

I submitted my binary through Application Loader, and the iTunes Connect status now says Invalid Binary. What does that mean?

Invalid Binary appears when a binary is received through Application Loader, has been processed, but is invalid. Examples of an invalid binary include:

- Your binary icon does not meet Apple's requirements
- You placed the payload directory at the wrong level in the .app wrapper
- You attempted to use a non-increasing CFBundleVersion

Apple sends an email to all Admin and Technical iTunes Connect users detailing the reason for your Invalid Binary. To clear the error, you must log in to iTunes Connect, locate your application, and click "Ready to Upload Binary" to reset your application's status to "Waiting for Upload". You can then redeliver your corrected binary through Application Loader.

How can I check the history and status of packages?

You can check the history and status of your packages before and after you deliver them to iTunes.

To check the history of your In-App Purchase before delivery:

- Choose File > Package History for your In-App Purchase in the Application Loader window, and then click Search.

To check the status of your In-App Purchase after delivery:

- Choose Window > Lookup Status in the Application Loader window, type the In-App Purchase SKU number, and then click Search.

Application Loader displays the current state and date created for the package.

I don't see any activities in the “Choose an Activity” window. What should I do?

If the “Choose an Activity” window does not display any activities, you do not have an active contract with iTunes. You must complete the contract process before you can deliver packages to iTunes and submit application binary files to the App Store. For more information, see the Contracts, Tax and Banking module in iTunes Connect at http://www.apple.com/itunes/go/itunesconnect/contracts.

How do I import a screenshot for my In-App Purchase?

Using Application Loader, you can import one or more screenshots at a time for your In-App Purchases. Each In-App Purchase can have only one screenshot associated with it. To import screenshots, do one of the following in the In-App Purchases > Info pane:

- Drag a screenshot to the “Screenshot for Review” box, or click Choose and select a screenshot file.
- Click Import Screenshots, click Yes, select the screenshot files your want to import or select an entire directory, and then click Open.

When you create a package using “New from file”, you can import multiple In-App Purchases at once and therefore import multiple screenshots for these In-App Purchases at once. You can choose the images individually or select an entire directory.

Notes:

- Screenshots must be a 320 x 460, 320 x 480, 480 x 320, or 480 x 300 pixel TIFF, JPEG, or PNG file in RGB Color mode.
- When importing multiple screenshots, screenshot filenames must match the In-App Purchase product identifiers (for example, *product_identifier*.png). For example, if a product identifier is com.cyberinteractive.touchfighter.missiles.fifty the matching screenshot filename must be com.cyberinteractive.touchfighter.missiles.fifty.png.
- Do not scale up artwork if it is smaller than the required size. If you are scanning your screenshot be sure to scan at the highest resolution, or at least 320 x 460 pixels. Screenshots must be at least 72 dots per inch (dpi). Do not submit art that contains an embedded color profile, back cover, or thumbnail art.

Can I sort information within Application Loader windows?

Many panes in Application Loader display information in lists, allowing you to sort the information by column heading. For example, in the In-App Purchases > Info pane, you can sort In-App Purchases by reference name, product identifier, type, price tier, or cleared for sale. To change the order in which Application Loader displays In-App Purchases in the list and sort In-App Purchases by product identifier, click the Product ID column heading. To reverse the sort order, click the column heading again.

Application Loader indicates the current sort order by highlighting the column and displaying ▼ in the column heading.

You can rearrange the order of the columns by dragging the column headers to the left or right. To adjust the width of the columns, drag the separator lines that are between the column headers.

Can I save a preliminary version of my In-App Purchase before sending it to iTunes?

If you're not ready to send your In-App Purchase to iTunes and want to finish later, you can save a preliminary version.

To save your In-App Purchase:

- Choose File > Save for your In-App Purchase in the Application Loader window.

  Application Loader saves your In-App Purchase as an Application Loader package using the application SKU number as the name and the .itmsp extension.

To reopen a saved package, open Application Loader, Click Open Package, and then double-click the package name.

By default, Application Loader saves packages in the Documents > Application Loader > Packages folder.

Note: By default, Application Loader automatically saves a package as *SKU*.itmsp every 15 seconds. To turn off autosaving or change the default autosave time, choose Application Loader > Preferences, click General, and then select "Autosave" or change the seconds.

What elements can I update in my In-App Purchase?

You cannot update all metadata for an In-App Purchase you previously delivered to iTunes. The following is a list of the details you can update with a redelivery:

- In-App Purchase details:

  Reference name
  Screenshot for review

- Rights & Pricing details:

  Cleared for sale
  Price tiers and dates

- Languages details:

  Language
  Display name
  Description

Note: Whether or not you can update screenshots and language details depends on the state of your In-App Purchase (for example, you cannot make updates if your application is in the "In Review" state in iTunes Connect).

Why is editing disabled for my In-App Purchase?

After you deliver your In-App Purchase to iTunes, Application Loader disables editing for fields you can no longer update.

To turn on editing for your In-App Purchase:

- Choose File > Enable Editing for your In-App Purchase in the Application Loader window.

Apple does not recommend enabling editing. If you enable editing and make updates to your In-App Purchase, you might not be able to deliver the package to iTunes. Use the enable editing feature when you are instructed to do so by Apple.

Glossary

Application Loader file A tab-delimited file (encoded in UTF-16 format) or spreadsheet file used to import In-App Purchase metadata details.

Application Loader package An Application Loader generated file with the .itmsp file extension. Application Loader generates a package file from your In-App Purchase metadata information, rights and pricing, and language details, and so on.

Existing (effective price tier date) Indicates that the price tier is currently in effect. The tier is already in existence and therefore you do not need to set a date to begin pricing at that tier.

In-App Purchases Purchases you can make within an iOS or Mac OS X application and managed by the iTunes Store (for example, billed to your iTunes Store account). For example, buying a specific digital book within an application on the iPad.

Invalid Binary A state in iTunes Connect indicating that your binary file was received through Application Loader but does not meet all requirements for upload.

iTunes Connect A suite of web-based tools you can use to manage your applications.

metadata Supplemental information about a media file type. For example, a file can contain information such as the name of the person that created the file, the length of the file, the title of the file, description, and so on.

None (end price tier date) Indicates that the price tier does not end on a specific date but instead remains in effect until you make a change.

Now (effective price tier date) Indicates that your price change takes effect is today.

screenshot for review A screenshot submitted with your Application Loader package for review purposes only. Screenshots are not displayed in the App Store or in the device Store.

SKU number A unique UTF-8 alphanumerical identifier for an application. The SKU is any alphanumeric sequence of letters and numbers you want to use to uniquely identify your application. You are free to create any string of UTF-8 letters and numbers, as long as it is unique to your provider account. You cannot edit your SKU number under any circumstance after submitting your metadata.

Waiting for Upload A state in iTunes Connect indicating that your binary file meets all requirements for upload. Your application must be in the "Waiting for Upload" state to be recognized in Application Loader as an eligible application for delivery.

Revision History

This table describes the changes to *Using Application Loader*.

Date	Notes
2011-08-17	This 2.4.1 release includes the following updates to Application Loader: improved user interface; In-App Purchase delivery; hardware accessories prototype testing; updated system requirements; send to Apple functionality; plus additional enhancements and bug fixes.
2010-09-17	Apple introduces Application Loader 1.3, a tool to help you submit your application binary files to the App Store.

Index