

Using Application Loader



Contents

Introduction 4

What's New 4

Organization of This Document 4

Getting Additional Information 5

Getting Started 6

What is Application Loader? 6

System and Network Requirements 6

Supported Formats 9

Installing Application Loader 10

Sign in to or out of Application Loader 11

 To sign in to Application Loader: 12

 To sign out of Application Loader: 13

Creating Your In-App Purchase 14

Creating a New In-App Purchase 15

Opening an Existing Package 16

Creating a Package from a File 17

Describing Your In-App Purchase 25

Entering In-App Purchase Information 25

Entering In-App Purchase Rights and Pricing 28

Entering In-App Purchase Languages 30

Entering In-App Purchase Hosted Content 32

Delivering Your In-App Purchase 34

Sending a Package to the App Store 34

Delivering Your App Binary 36

Uploading Your Application Binary Files with Application Loader 36

Uploading Your Application Binary Files with altool 38

Common Questions 39

What is Application Loader? 39

I don't think I have Application Loader currently installed. How can I get it?	39
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?	39
Which operating system is required to run Application Loader?	40
I uploaded my binary through Application Loader and the binary status is now Invalid Binary. What does that mean?	40
How do I check the history and status of packages?	40
I don't see any activities in the Template Chooser window. What should I do?	41
How do I import a screenshot for my In-App Purchase?	41
Can I sort information within Application Loader windows?	42
Can I save a preliminary version of my In-App Purchase?	42
What elements can I update in my In-App Purchase?	43
Why is editing disabled for my In-App Purchase?	43

Glossary 44

Revision History 45

Index 47

Introduction

This guide describes Application Loader 3.1, an Apple tool to help you prepare your applications for sale in the App Store. You can use Application Loader to create an Application Loader package containing the In-App Purchase metadata you want to add to your application and to upload your application binary files to the App Store.

Application Loader offers fast uploading, stable connections, and early validation warnings.

What's New

Application Loader 3.1 contains the following enhancements:

- Enhanced submission feedback. Direct links to Application Loader support articles to resolve submission issues. For details, see [“Delivering Your In-App Purchase”](#) (page 34) and [“Delivering Your App Binary”](#) (page 36).
- Command-line validation and upload. You can now use Application Loader’s command-line tool to validate your build before upload or to automate uploads of valid builds to the App Store. For details, see [“Delivering Your In-App Purchase”](#) (page 34).

Organization of This Document

This guide is organized as follows:

- [“Getting Started”](#) (page 6), 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 14), describes how to create an Application Loader package containing an In-App Purchase to submit to the App Store.
- [“Describing Your In-App Purchase”](#) (page 25), explains how to prepare your Application Loader package containing an In-App Purchase for delivery to the App Store by providing In-App Purchase metadata.
- [“Delivering Your In-App Purchase”](#) (page 34), provides details on how to deliver your Application Loader package containing an In-App Purchase to the App Store.

- “[Delivering Your App Binary](#)” (page 36), provides details on how to deliver your application binary files to the App Store as well as describes Application Loader’s command-line tool for validating and uploading your applications to the App Store.
- “[Common Questions](#)” (page 39), features some of the common questions you might have while using Application Loader.
- “[Glossary](#)” (page 44), 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 on 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 “Resources and Help” on iTunes Connect at <http://www.apple.com/itunes/go/itunesconnect/contactus>.

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 to create an Application Loader package containing the In-App Purchase metadata you want to add to your application and to upload your application binary files to the App Store.

Application Loader offers fast uploading, stable connections, and early validation warnings.

System and Network Requirements

Application Loader 3.1 requires a Mac, at least 512 MB RAM, and OS X v10.9 or later installed. Apple recommends at least 20 GB of available space and a broadband internet connection with an upload rate of 1MB/sec or faster.

Application Loader is integrated with the Xcode developer tools package. Therefore, 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	Notes
itmsdav.apple.com	17.152.19.125	443	n/a	DAV only
contentdelivery.itunes.apple.com	17.152.19.127	443	n/a	HTTPS*
idmsa.apple.com	17.151.2.14	443	n/a	HTTPS*
vgr101.apple.com	17.152.249.51	33001	33001-33500	Aspera only
vgr102.apple.com	17.152.249.52	33001	33001-33500	Aspera only

Server	IP Address	TCP Port	UDP Port	Notes
vgr103.apple.com	17.152.249.53	33001	33001-33500	Aspera only
vgr104.apple.com	17.152.249.54	33001	33001-33500	Aspera only
vgr105.apple.com	17.152.249.55	33001	33001-33500	Aspera only
vgr106.apple.com	17.152.249.56	33001	33001-33500	Aspera only
vgr107.apple.com	17.152.249.57	33001	33001-33500	Aspera only
vgr108.apple.com	17.152.249.58	33001	33001-33500	Aspera only
vgr301.apple.com	17.172.190.141	33001	33001-33500	Aspera only
vgr302.apple.com	17.172.190.142	33001	33001-33500	Aspera only
vgr303.apple.com	17.172.190.143	33001	33001-33500	Aspera only
vgr304.apple.com	17.172.190.144	33001	33001-33500	Aspera only
vgr305.apple.com	17.172.190.145	33001	33001-33500	Aspera only
vgr306.apple.com	17.172.190.146	33001	33001-33500	Aspera only
vgr307.apple.com	17.172.190.147	33001	33001-33500	Aspera only
vgr308.apple.com	17.172.190.148	33001	33001-33500	Aspera only
sgr201.apple.com	17.152.249.101	44001	44001-44500	Signiant only
sgr202.apple.com	17.152.249.102	44001	44001-44500	Signiant only
sgr203.apple.com	17.152.249.103	44001	44001-44500	Signiant only
sgr204.apple.com	17.152.249.104	44001	44001-44500	Signiant only
sgr205.apple.com	17.152.249.105	44001	44001-44500	Signiant only
sgr206.apple.com	17.152.249.106	44001	44001-44500	Signiant only
sgr207.apple.com	17.152.249.107	44001	44001-44500	Signiant only
sgr208.apple.com	17.152.249.108	44001	44001-44500	Signiant only
sgr401.apple.com	17.172.190.101	44001	44001-44500	Signiant only
sgr402.apple.com	17.172.190.102	44001	44001-44500	Signiant only

Server	IP Address	TCP Port	UDP Port	Notes
sgr403.apple.com	17.172.190.103	44001	44001-44500	Signiant only
sgr404.apple.com	17.172.190.104	44001	44001-44500	Signiant only
sgr405.apple.com	17.172.190.105	44001	44001-44500	Signiant only
sgr406.apple.com	17.172.190.106	44001	44001-44500	Signiant only
sgr407.apple.com	17.172.190.107	44001	44001-44500	Signiant only
sgr408.apple.com	17.172.190.108	44001	44001-44500	Signiant only
vgr501.apple.com	17.110.248.141	33001	33001-33500	Aspera only
vgr502.apple.com	17.110.248.142	33001	33001-33500	Aspera only
vgr503.apple.com	17.110.248.143	33001	33001-33500	Aspera only
vgr504.apple.com	17.110.248.144	33001	33001-33500	Aspera only
vgr505.apple.com	17.110.248.145	33001	33001-33500	Aspera only
vgr506.apple.com	17.110.248.146	33001	33001-33500	Aspera only
vgr507.apple.com	17.110.248.147	33001	33001-33500	Aspera only
vgr508.apple.com	17.110.248.148	33001	33001-33500	Aspera only
sgr601.apple.com	17.110.248.101	44001	44001-44500	Signiant only
sgr602.apple.com	17.110.248.102	44001	44001-44500	Signiant only
sgr603.apple.com	17.110.248.103	44001	44001-44500	Signiant only
sgr604.apple.com	17.110.248.104	44001	44001-44500	Signiant only
sgr605.apple.com	17.110.248.105	44001	44001-44500	Signiant only
sgr606.apple.com	17.110.248.106	44001	44001-44500	Signiant only
sgr607.apple.com	17.110.248.107	44001	44001-44500	Signiant only
sgr608.apple.com	17.110.248.108	44001	44001-44500	Signiant only

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 the following file formats:

- IPA files (.ipa) for delivering your iOS apps.
- PKG (.pkg) package files for delivering your OS X apps and In-App Purchases. For information about how to use productbuild to create a package file, see the productbuild manual page in the *Mac Developer Library* at https://developer.apple.com/library/mac/#documentation/Darwin/Reference/ManPages/man1/productbuild.1.html%23//apple_ref/doc/man/1/productbuild.
- ZIP files (.zip) produced by Finder (passing a code sign check upon upload) for delivering your application binary. For specific details, refer to the *Developer Reference Library* in the Apple Dev Center for creating properly signed apps.

Screenshots should be a clear picture of your In-App Purchase in action, JPEG or PNG file in RGB Color mode, and the following dimensions:

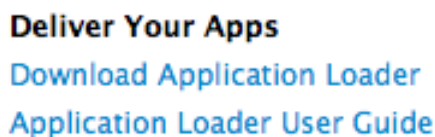
- For OS X: 1280 x 800, 1440 x 900, 2560 x 1600, or 2880 x 1800 pixels
- For iOS: 960 x 640, 960 x 600, 640 x 960, 640 x 920, 1024 x 768, 1024 x 748, 768 x 1024, 768 x 1004, 2048 x 1536, 2048 x 1496, 1536 x 2048, or 1536 x 2008 pixels

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

You can install and run Application Loader 3.1 on the OS X operating system. To install Application Loader:

1. Click My Apps in the [iTunes Connect](#) home page.
iTunes Connect displays My Apps with installation and documentation downloads.
2. Click the Download Application Loader link to download the self-extracting installation package (ApplicationLoader_3.1.dmg).



Deliver Your Apps
[Download Application Loader](#)
[Application Loader User Guide](#)

3. Double-click the installation package (ApplicationLoader_3.1.dmg).
4. Double-click the installation package (ApplicationLoader.pkg) and follow the instructions that appear.

Note: When installing Application Loader on the Mac, before you can install the software, you must log in to your computer as a member of the administrator group.

5. Read the installation introduction, then click Continue.



6. Click Install to perform the standard installation.

The installation package installs files in your Applications folder.

Type your administrator password and click OK, if prompted.

7. Click Close.

Open the Application Loader application by double-clicking the “Application Loader.app” icon in the Applications folder, or the folder you specified during a custom install. You are now ready to begin using Application Loader. For more information, see [“Sign in to or out of Application Loader”](#) (page 11) later in this section.

Note: To access Application Loader from the Xcode developer tools package, choose Xcode > Open Developer Tool > Application Loader. 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.

Sign in to or out of Application Loader

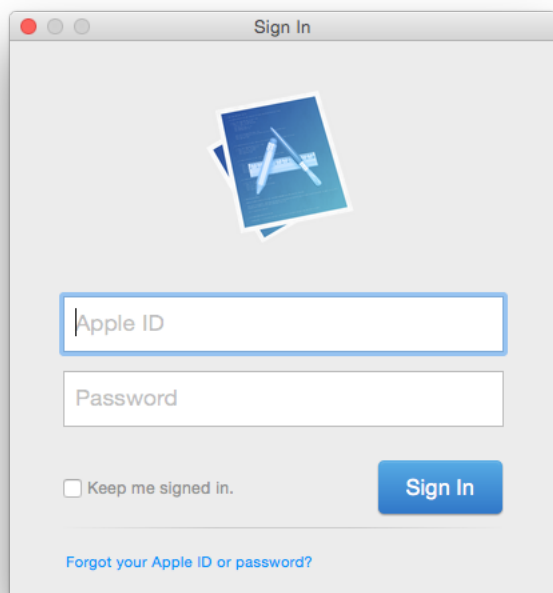
Before you can use Application Loader to create an Application Loader package and to upload your application binary files to the App Store, you must sign in to Application Loader using your [iTunes Connect](#) sign-in information. To prevent anyone from accessing your Application Loader account while you are away from your computer, you can sign out of your account.

Note: Application Loader displays an active user button containing the Apple ID for the currently signed in user. Click the button to display the Sign In window to change or update your Apple ID and password.

To sign in to Application Loader:

1. Double-click the Application Loader.app icon in the Applications folder or, to access Application Loader from the Xcode developer tools package, choose Xcode > Open Developer Tool > Application Loader.
2. Read the Application Loader Software License Agreement, then click Agree.

Application Loader opens the Sign In window.



3. Type your iTunes Connect sign-in information (Apple ID and password) in the Sign In window.
4. Click "Keep me signed in" to stay signed in when you quit Application Loader.
5. Click Sign In.

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

Application Loader displays the Template Chooser window so you can begin using Application Loader. For more information, see ["Creating Your In-App Purchase"](#) (page 14) later in this guide.

If you change your iTunes Connect password, you must use the Sign In window to reset your Apple ID and password in Application Loader. To reset your Apple ID and password, choose Window > Sign In As, then repeat steps 3 through 6 above.

To sign out of Application Loader:

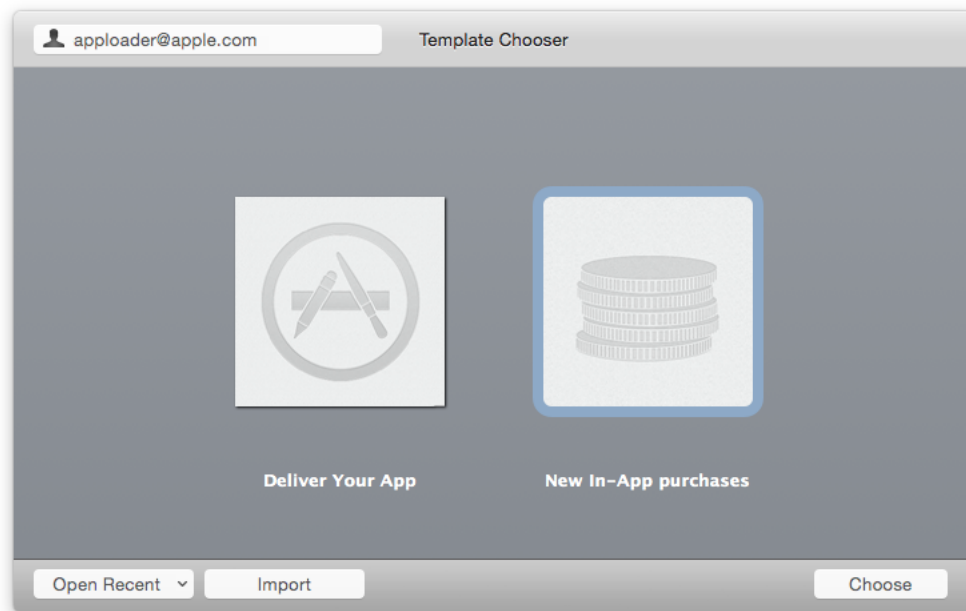
Do one of the following:

- Choose Window > Sign In As, then sign in using new iTunes Connect sign-in information.
- Deselect “Keep me signed in” in the Sign In window, then quit Application Loader or, to change accounts, click the active user button, then sign in using new iTunes Connect sign-in information.

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 screenshot displays the Application Loader Template Chooser window with the “New In-App purchases” option selected.



If you do not see any of the above activities in the Application Loader Template Chooser window, you might not have configured Application Loader delivery. Click the active user button to display the Sign In window again.

Creating a New In-App Purchase

You can create a package containing the In-App Purchase you want to submit to the App Store by creating a new In-App Purchase. To add metadata using a new In-App Purchase:

1. Click “New In-App purchases” in the Template Chooser window, then click Choose.
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, and SKU information.
2. Select the application where you want to add In-App Purchases, then click Manage.
Application Loader displays a *SKU.itmsp* or *Untitled.itmsp* Application Loader window with the In-App Purchases pane selected.
3. Use the In-App Purchases pane to add new In-App Purchases with metadata information, rights and pricing, languages, and content hosting. For more information, see [“Describing Your In-App Purchase”](#) (page 25).

When you are sure the In-App Purchase details are complete and correct, you can send the package to the App Store. For more information, see [“Delivering Your In-App Purchase”](#) (page 34).

Opening an Existing Package

You can create a package containing the In-App Purchase you want to submit to the App Store by opening a previously saved [Application Loader package](#) and adding and editing metadata. To add metadata from an existing package:

1. Do one of the following in the Template Chooser window:
 - Click Open Recent, then choose the local package file from the pop-up menu.
 - Choose File > Open, then choose the local package file from the Open window.

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

2. Use the In-App Purchases pane to add new In-App Purchases and add and edit metadata information, rights and pricing, languages, and content hosting. For more information, see [“Describing Your In-App Purchase”](#) (page 25).

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

Note: If you previously delivered the package to the App Store, 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 43)

When you are sure the In-App Purchase details are complete and correct, you can send the package to the App Store. For more information, see [“Delivering Your In-App Purchase”](#) (page 34).

Creating a Package from a File

You can create a package containing the In-App Purchase you want to submit to the App Store 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. To get a copy of the template, choose File > Open App Import Template in Application Loader. Application Loader opens Finder and creates a temporary AppFileImportTemplates folder containing the AppMassImportTemplate.txt template. To always have an original version of the template file, be sure to make a copy of the template before editing.

To import metadata details from a file:

1. Click Import, select the filename containing the metadata details you want to import into the new package, then click Open.

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

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

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 the matching screenshot filename must be com.cyberinteractive.touchfighter.png.

3. Click Yes to import multiple hosted contents at once, if applicable.

You can import multiple In-App Purchases at once and therefore import multiple hosted contents for these In-App Purchases at once. You can choose packages individually or select an entire directory. Package filenames must match the In-App Purchase product identifiers (for example, *product_identifier.png*). For example, if a product identifier is com.acme.inappppurchase1 the matching package filename must be com.acme.inappppurchase1.pkg.

Application Loader imports the details from the Application Loader file and displays a *SKU .itmsp* or *Unsaved Application Loader Document .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 window, with the In-App Purchases pane selected, for each unique SKU.

4. Use the In-App Purchases pane to add new In-App Purchases and add and edit metadata information, rights and pricing, languages, and content hosting. For more information, see [“Describing Your In-App Purchase”](#) (page 25).

When you are sure the In-App Purchase details are complete and correct, you can send the package to the App Store. For more information, see [“Delivering Your In-App Purchase”](#) (page 34).

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
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
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 on iTunes Connect and in sales and trends reports.	30 missiles
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

Name	Description	Example
Wholesale Price Tier	<p>A retail price tier for the current In-App Purchase.</p> <p>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.</p>	3
Displayed Name <i>language</i>	<p>A user-visible display name (from 2 to 75 characters) where <i>language</i> is a supported localized language. You can add a new column for each localized language display name.</p> <p>Note: When specifying the displayed name, replace <i>language</i> with either the language name (for example, English), localized version of the language (for example, French or français), or the locale code (for example, en-US). Leave the “Displayed Name” column title text in English. For example, “Displayed Name English” and “Displayed Name French”.</p>	30 missiles 30 missiles
Description <i>language</i>	<p>A user-visible description (from 10 to 255 characters) where <i>language</i> is a supported localized language. You can add a new column for each localized language description.</p> <p>Note: When specifying the description, replace <i>language</i> with either the language name (for example, French), localized version of the language (for example, French or français), or the locale code (for example, fr-FR). Leave the “Description” column title text in English. For example, “Description English” and “Description French”.</p>	Buy thirty extra missiles to help you on your way. Acheter trente missiles supplémentaires pour vous aider sur votre chemin.

Name	Description	Example
Screenshot Path	<p>The absolute or relative pathname and filename, without spaces, for the associated screenshot for review.</p> <p>Screenshots should be a clear picture of your In-App Purchase in action, JPEG or PNG file in RGB Color mode, and the following dimensions:</p> <ul style="list-style-type: none">• For OS X: 1280 x 800, 1440 x 900, 2560 x 1600, or 2880 x 1800 pixels• For iOS: 960 x 640, 960 x 600, 640 x 960, 640 x 920, 1024 x 768, 1024 x 748, 768 x 1024, 768 x 1004, 2048 x 1536, 2048 x 1496, 1536 x 2048, or 1536 x 2008 pixels	<code>../SC/com.cyberinteractive.touchfighter.jpg</code>
Effective Date	<p>The date when the new price tier takes effect for the current In-App Purchase.</p> <p>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.</p>	Jun 6, 2011

Name	Description	Example
End Date	<p>The date when the new price tier reverts to the previously specified price tier for the current In-App Purchase.</p> <p>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.</p>	Jun 10, 2011
Review Notes	Additional information (up to 4000 bytes) about your In-App Purchase.	Click the Touchfighter "Store" button in the top right corner to purchase these missiles. After purchase, you can select the missiles from the Weapons panel.

Name	Description	Example
Hosted Content Path	<p>The absolute or relative pathname and filename, without spaces, for the non-consumable In-App Purchase content package file you want to host on Apple servers.</p> <p>Hosted content can be additional resources, but not executables. Examples include filters on a camera app, a level map for a game, add-on videos, images, or a magazine issue. Customers purchase non-consumable products only once. Services that do not expire or decrease with use, such as a new race track for a game app, are usually implemented as non-consumables. The non-consumable product is provided to all devices associated with the customer's iTunes account, and available for iOS and OS X applications.</p> <p>Content package files must have a .pkg file extension.</p>	../HC/com.cyberinteractive.touchfighter.pkg

Name	Description	Example
	<p>When sending a content package for your non-consumable In-App Purchase for Apple hosting, keep the following in mind:</p> <ul style="list-style-type: none">• The size of the content package cannot exceed 2GB.• The content package must not contain any executable code or symlinks that point outside the content directory.• You can deliver a new package once your In-App Purchase is live. For your new content to go live, you must submit it for review and approval on iTunes Connect.• To remove files, if your In-App Purchase is not live, uncheck "Host Content with Apple" and resend your package. You cannot remove files once you deliver your package and your In-App Purchase is live.	

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 1,000 In-App Purchases per application.
- You can exclude any column data, as long as you leave the column and 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, select the SKU column in your spreadsheet, choose Format > Cells, then select Text for the category. Now you can properly enter your SKU numbers in the cells.

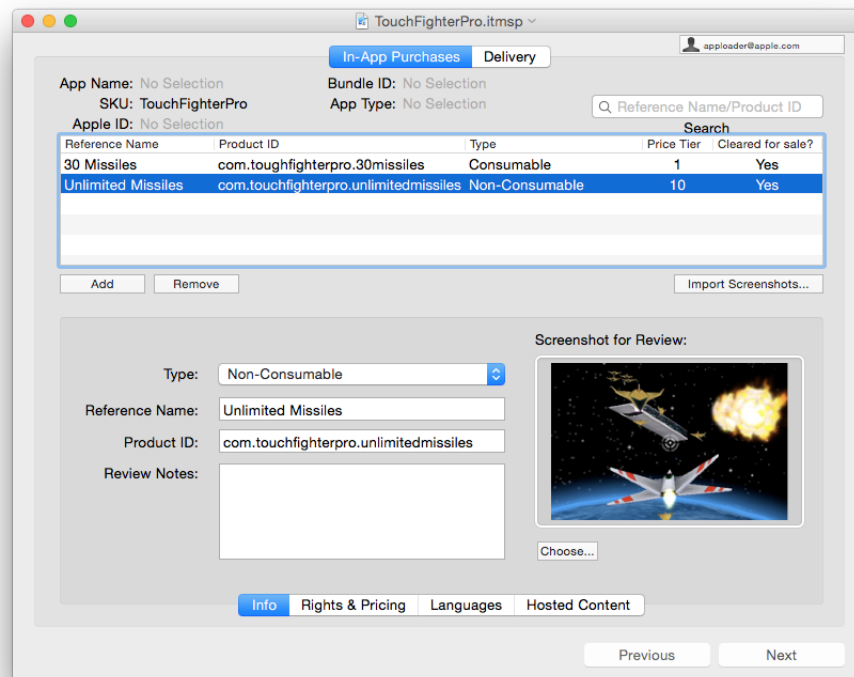
Describing Your In-App Purchase

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 the App Store, you must use the In-App Purchases pane of Application Loader to prepare the In-App Purchase by describing the metadata details you want to submit, including information, rights and pricing, languages, and content hosting.

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



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 1,000 In-App Purchases per application.

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

3. Choose the type of In-App Purchase you want to add from the Type pop-up menu.
 - **Consumable** (available for iOS and 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 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 on 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.
 - **Review Notes:** Type up to 4000 bytes of additional information about your In-App Purchase.
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 should be a clear picture of your In-App Purchase in action, JPEG or PNG file in RGB Color mode, and the following dimensions:

- For OS X: 1280 x 800, 1440 x 900, 2560 x 1600, or 2880 x 1800 pixels
- For iOS: 960 x 640, 960 x 600, 640 x 960, 640 x 920, 1024 x 768, 1024 x 748, 768 x 1024, 768 x 1004, 2048 x 1536, 2048 x 1496, 1536 x 2048, or 1536 x 2008 pixels

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* the matching screenshot filename must be *com.cyberinteractive.touchfighter.png*.

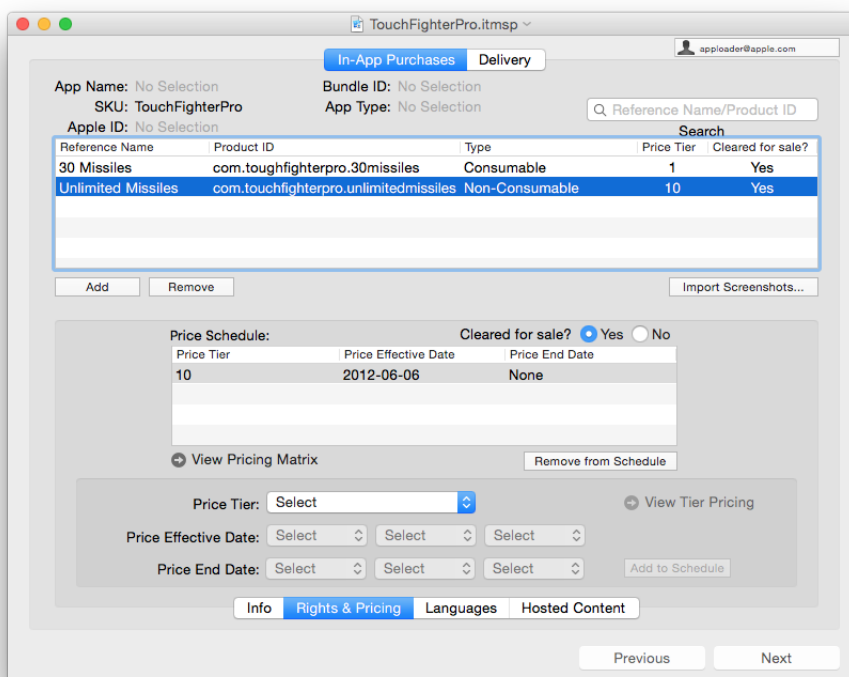
Do not scale up artwork if it is smaller than the required size. If you are scanning your screenshot be sure to scan at the highest resolution. Screenshots must be at least 72 dots per inch (dpi). Do not submit art that contains an embedded color profile, back cover, or thumbnail art.

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 28).

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 an end date. Application Loader provides pricing matrix and pricing tier tables 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.



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 1,000 In-App Purchases per application.
Use the search field to filter In-App Purchases in the table by Reference Name and Product ID 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 a retail price tier for the current In-App Purchase from the Price Tier pop-up menu.

5. 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.
6. 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.
7. 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.
Click "Remove from Schedule" to remove an existing price tier from the Price Schedule table and the current In-App Purchase.
8. Repeat steps 4 through 7 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.
Select a price tier and click View Tier Pricing to display pricing tier information for the currently selected tier.
9. Repeat steps 2 through 8 for each price tier you want to add to each In-App Purchase in the current application.
10. Click Next or Languages at the bottom of the In-App Purchases pane. For more information, see ["Entering In-App Purchase Languages"](#) (page 30).

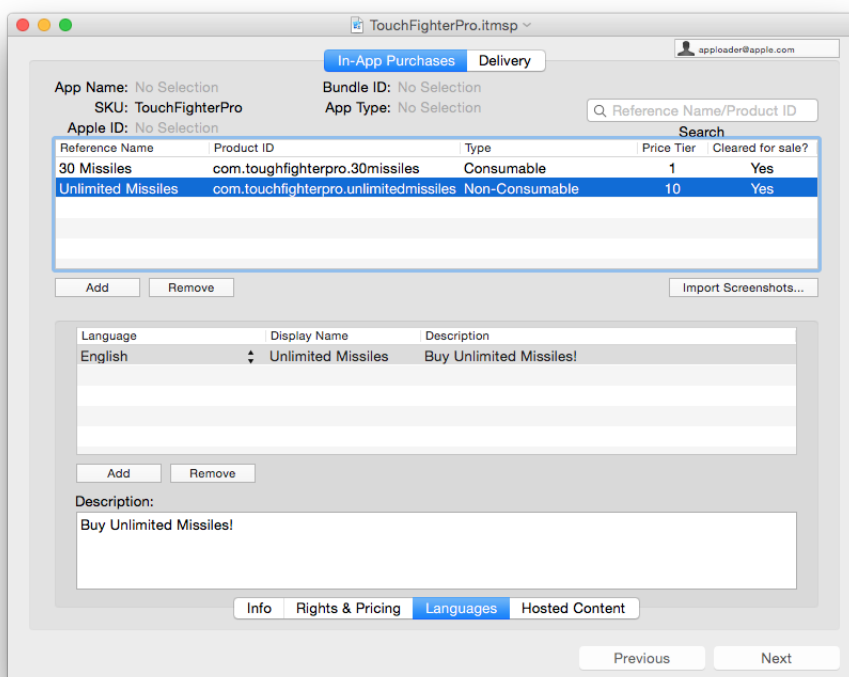
Notes:

- You can set up as many price tier changes as you want and your application will change prices on your effective dates.
- To create a default wholesale price tier, choose None from the Price Effective Date pop-up menus and None from the Price End Date pop-up menus. Application Loader displays the default wholesale price tier first in the Price Tier column. If a default wholesale price tier already exists, Application Loader replaces it with the newly created price tier.
- 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.



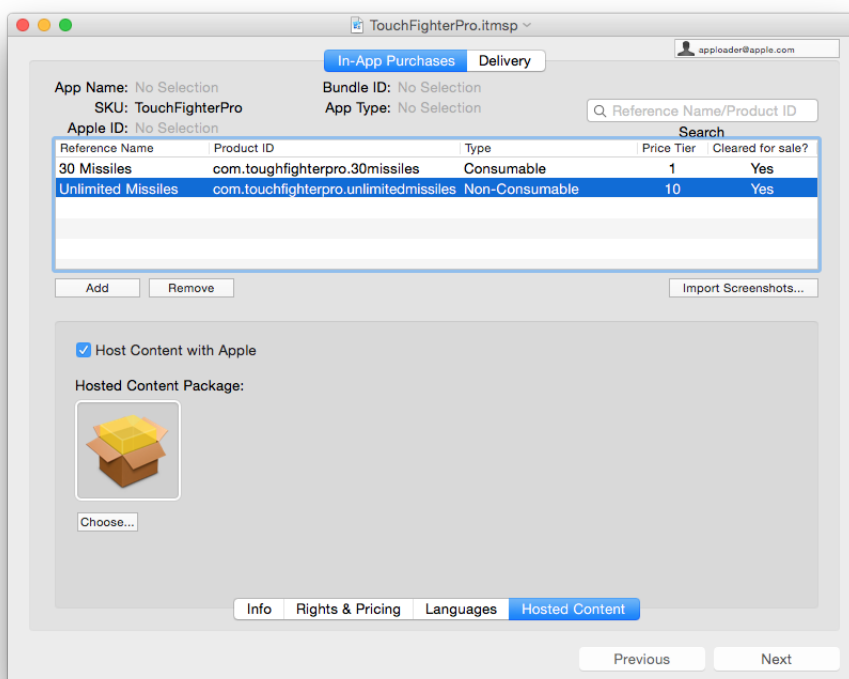
2. Select an existing In-App Purchase or click Add to add a new In-App Purchase for the current application. Click Remove to delete the selected In-App Purchase. Application Loader removes the In-App Purchase from the table but does not delete it from iTunes Connect. Application Loader supports a maximum of 1,000 In-App Purchases per application. Use the search field to filter In-App Purchases in the table by Reference Name and Product ID information.
3. Click Add to add a language for the current In-App Purchase, 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, 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, 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 or Hosted Content at the bottom of the In-App Purchases pane. For more information, see [“Entering In-App Purchase Hosted Content”](#) (page 32).

Entering In-App Purchase Hosted Content

Use the In-App Purchases > Hosted Content pane to submit a content package for your non-consumable In-App Purchase and request that Apple host the content on Apple servers. Non-consumable In-App Purchases can be hosted by Apple so you do not need to maintain your own servers. The product type for hosted content must be non-consumable. Hosted content can be additional resources, but not executables. Examples include filters on a camera app, a level map for a game, add-on videos, images, or a magazine issue. Customers purchase non-consumable products only once. Services that do not expire or decrease with use, such as a new race track for a game app, are usually implemented as non-consumables. The non-consumable product is provided to all devices associated with the customer's iTunes account, and available for iOS and OS X applications. If you have the correct contracts with iTunes, you can create a content package for your non-consumable In-App Purchase and send it to Apple for hosting. To provide In-App Purchase hosted content details:

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



2. Click “Host Content with Apple” to indicate that your In-App Purchase is hosted by Apple, not on your own servers.

Currently, you can only host non-consumable In-App Purchases.

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

3. Drag a package to the Hosted Content Package box, or click Choose and select a package file for the In-App Purchase. To replace a previously added file, drag a new file to the Hosted Content Package box or click Choose to select a new file, or choose Edit > Delete to delete the file and then add a new file.

Content package filenames cannot contain spaces and must have a .pkg file extension.

4. Click Next at the bottom of the In-App Purchases pane. For more information, see [“Delivering Your In-App Purchase”](#) (page 34).

When sending a content package for your non-consumable In-App Purchase for Apple hosting, keep the following in mind:

- The size of the content package cannot exceed 2GB.
- The content package must not contain any executable code or symlinks that point outside the content directory.
- You can deliver a new package once your In-App Purchase is live. For your new content to go live, you must submit it for review and approval on iTunes Connect.
- To remove files, if your In-App Purchase is not live, uncheck “Host Content with Apple” and resend your package. You cannot remove files once you deliver your package and your In-App Purchase is live.

Delivering Your In-App Purchase

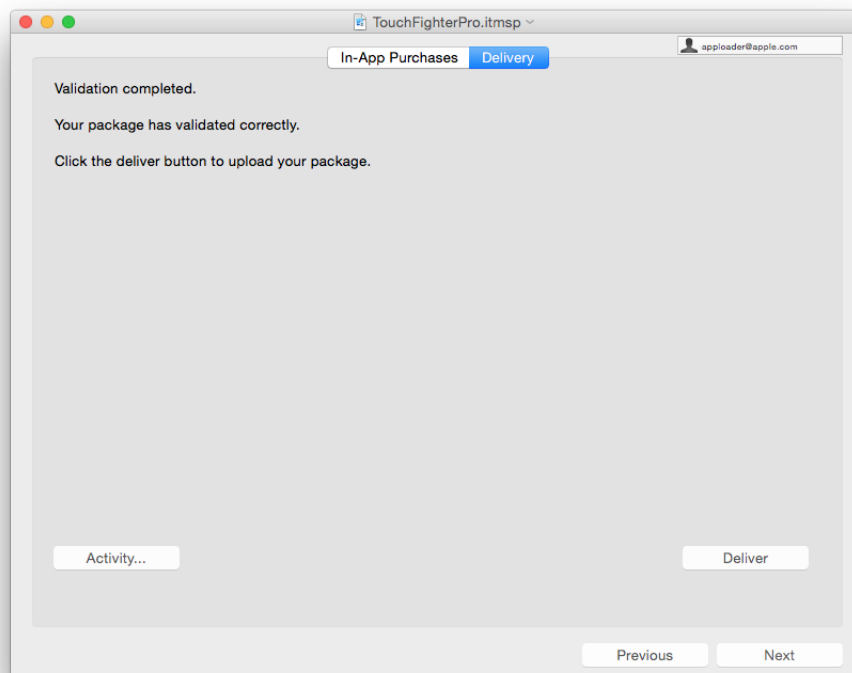
The final step in adding your In-App Purchase to your application is to deliver the [Application Loader package](#) to the App Store. Once you have created the package and entered all your In-App Purchase metadata details, you can deliver the package and all your prepared files as an Application Loader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

Sending a Package to the App Store

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 the App Store. To deliver a package:

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

If applicable, specify a name for the package, then click Save.



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 the App Store.

Click  to stop the transfer and keep the window open.

Fix any delivery warnings or errors, then deliver your app again. Click View Log to review detailed information about any delivery warnings or errors.

For information about delivery errors, click FAQs to display Frequently Asked Questions on iTunes Connect.

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 and displays a green checkmark 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.
- 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 the App Store. 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 the App Store.
- To modify your delivery settings, choose Application Loader > Preferences, click Advanced, 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 “Resources and Help” on iTunes Connect at <http://www.apple.com/itunes/go/itunesconnect/contactus>.

Delivering Your App Binary

When you have finished uploading your assets and completed the application submission process on iTunes Connect, you can deliver your application binary files to the App Store.

This section describes how to upload your application binary files to the App Store. In addition, this section describes altool, Application Loader's command-line tool for validating and uploading your application binary files to the App Store.

Uploading Your Application Binary Files with Application Loader

You can use Application Loader to upload your application binary files to the App Store. To upload your application binary files:

1. Click Deliver Your App in the Template Chooser window, then click Choose.
2. Select your application binary file from the Deliver Your App window, then click Open.


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



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 Next.

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

Click  to stop the transfer and keep the window open.

Fix any delivery warnings or errors, then deliver your app again. Click View Log to review detailed information about about any delivery warnings or errors.

Use [My Apps](#) on 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.

Uploading Your Application Binary Files with altool

You can use altool, Application Loader's command-line tool for validating and uploading your application binary files to the App Store.

To validate your build before upload or to automate uploads of valid builds to the App Store, you can include altool in your continuous integration systems. altool is located in the Application Loader.app/Contents/Frameworks/ITunesSoftwareService.framework/Versions/A/Support/ folder.

To run altool, specify one of the following at the command-line:

```
$ altool --validate-app -f file -u username [-p password] [--output-format xml]
$ altool --upload-app -f file -u username [-p password] [--output-format xml]
```

Where:

--validate-app	Specifies you want to validate the specified application.
--upload-app	Specifies you want to upload the specified application.
-f file	Specifies the path and filename for the application you are validating or uploading.
-u username	Specifies your username.
-p password	Specifies your user password.
--output-format [xml normal]	Specifies that you want Application Loader to return output in structured XML format or unstructured text format. By default, Application Loader returns output information in text format.

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 to create an Application Loader package containing the In-App Purchase metadata you want to add to your application and to upload 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.

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

You can download Application Loader 3.1 from iTunes Connect or, when you install the Xcode developer tools package, the installation package automatically installs Application Loader 3.1 files. To open the Application Loader application, double-click the “Application Loader.app” icon in the Applications folder or, to access Application Loader from the Xcode developer tools package, choose Xcode > Open Developer Tool > Application Loader.

For more information, see [“Installing 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 Sign In window. To access the Sign In window, choose Window > Sign In As in Application Loader.

Which operating system is required to run Application Loader?

Application Loader 3.1 requires a Mac, at least 512 MB RAM, and OS X v10.9 or later installed. Apple recommends at least 20 GB of available space and a broadband internet connection with an upload rate of 1 MB/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.

I uploaded my binary through Application Loader and the binary status is now 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. Apple recommends uploading a new binary.

How do I check the history and status of packages?

You can check the history and status of packages after you deliver them to the App Store.

To check package history:

- Choose File > Package History for your In-App Purchase in the Application Loader window, type the In-App Purchase SKU or Vendor ID number, then click Search.

To check package status:

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

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

I don't see any activities in the Template Chooser window. What should I do?

If the Template Chooser 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 the App Store and upload application binary files to the App Store. For more information, see Contracts, Tax and Banking in [iTunes Connect](http://www.apple.com/itunes/go/itunesconnect/contracts) 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 you want to import or select an entire directory, then click Open.

When you create a package by importing a 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 should be a clear picture of your In-App Purchase in action, JPEG or PNG file in RGB Color mode, and the following dimensions:
 - For OS X: 1280 x 800, 1440 x 900, 2560 x 1600, or 2880 x 1800 pixels
 - For iOS: 960 x 640, 960 x 600, 640 x 960, 640 x 920, 1024 x 768, 1024 x 748, 768 x 1024, 768 x 1004, 2048 x 1536, 2048 x 1496, 1536 x 2048, or 1536 x 2008 pixels
- 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 the matching screenshot filename must be com.cyberinteractive.touchfighter.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. 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?

If you're not ready to send your In-App Purchase to the App Store 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 .itmzp extension.

To reopen a saved package, open Application Loader, choose File > Open, then double-click the package name.

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

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

You cannot update all metadata for an In-App Purchase you previously delivered to the App Store. 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 on iTunes Connect).

Why is editing disabled for my In-App Purchase?

After you deliver your In-App Purchase to the App Store, 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 the App Store. 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, language, and content hosting 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.

hosted content Additional resources, but not executables. For example, filters on a camera app, a level map for a game, add-on videos, images, or a magazine issue. The product type for hosted content must be non-consumable. Customers purchase non-consumable products only once. Services that do not expire or decrease with use, such as a new race track for a game app, are usually implemented as non-consumables. The non-consumable product is provided to all devices associated with the customer's iTunes account, and available for iOS and OS X applications.

In-App Purchases Purchases you can make within an iOS or 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 on 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 on a user's device.

SKU number A unique UTF-8 alphanumeric 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.

Revision History

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

Date	Notes
2015-04-06	This 3.1 release includes the following updates to Application Loader: enhanced submission feedback; command-line validation and upload; plus additional enhancements and bug fixes.
2014-08-13	This 3.0 release includes the following updates to Application Loader: improved user interface; simplified Deliver Your App process; plus additional enhancements and bug fixes.
2014-07-16	This 2.9.1 release includes the following updates to Application Loader: updated system requirements; plus additional enhancements and bug fixes.
2013-07-28	This 2.9 release includes the following updates to Application Loader: enhanced rights and pricing, including pricing tier information for the currently selected tier; updated file format support; additional import fields when creating a package from a file, including Hosted Content Path; localization updates; enhanced validation and error feedback; plus additional enhancements and bug fixes.
2012-06-12	This 2.7 release includes the following updates to Application Loader: content hosting; import templates; plus additional enhancements and bug fixes.
2012-02-20	This 2.5.1 release includes the following updates to Application Loader: iTunes Connect installer; enhanced rights and pricing; improved user interface; additional In-App Purchases support; plus additional enhancements and bug fixes.

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

A

activities, none [41](#)
alttool [38](#)
app binary, uploading to the App Store [36, 38](#)
App Store, uploading app binary [36](#)
Application Loader file, definition [44](#)
Application Loader package, definition [44](#)
Application Loader
 command-line [38](#)
 contacting [5](#)
 getting started [6, 11](#)
 help [5](#)
 installing [10, 39](#)
 introduction [4, 6, 39](#)
 overview [6, 39](#)
 password [39](#)
 sign in [11, 39](#)
 sign out [11](#)
 support [5](#)
 what's new [4](#)

B

background activity [37](#)

C

cleared for sale, In-App Purchase [28](#)
columns
 rearrange [42](#)
 sorting [42](#)
command-line, alttool [38](#)
consumable, In-App Purchase [26](#)
content hosting, In-App Purchase [32](#)

D

delivering, In-App Purchase [34](#)
description localized, In-App Purchase [31](#)
display name localized, In-App Purchase [31](#)

E

elements, editable in In-App Purchase [43](#)
Existing (effective price tier date), definition [44](#)

F

fields, import [18](#)
file
 creating In-App Purchase [17](#)
 formats [9](#)
 invalid [40](#)
filtering [26, 28, 30, 32](#)
formats, supported [9](#)

H

help [5](#)
hosted content package, In-App Purchase [32, 33](#)
hosted content, definition [44](#)

I

import fields [18](#)
In-App Purchase
 cleared for sale [28](#)
 consumable type [26](#)
 creating [14](#)
 creating from existing package [16](#)
 creating from file [17](#)

- creating new 15
- default wholesale price tier 29
- delivering to iTunes 34
- describing 25
- details 25
- editable elements 43
- editing 43
- hosted content 32
- hosted content package 33
- information 25
- languages 30
- localized description 31
- localized display name 31
- non-consumable type 26
- non-renewing subscription type 26
- price effective date 29
- price end date 29
- pricing 28
- product identifier 26
- reference name 26
- retail price tier 28
- review notes 26
- rights 28
- saving 42
- screenshots 26
- sending to iTunes 34
- In-App Purchases, definition 44
- installing Application Loader 10, 39
- Invalid Binary, definition 44
- invalid file 40
- IP addresses 6
- iTunes Connect
 - definition 44
 - password 39
 - sign in 39
- iTunes
 - delivering In-App Purchase 34
 - sending In-App Purchase 34

L

- languages, In-App Purchase 30

M

- metadata, definition 44

N

- non-consumable, In-App Purchase 26
- non-renewing subscription, In-App Purchase 26
- None (end price tier date), definition 44
- Now (effective price tier date), definition 44

O

- operating system 40

P

- package
 - history 40
 - status 40
- password
 - Application Loader 39
 - iTunes Connect 39
- ports 6
- price effective date, In-App Purchase 29
- price end date, In-App Purchase 29
- pricing, In-App Purchase 28
- product identifier, In-App Purchase 26

R

- reference name, In-App Purchase 26
- requirements, system 6, 40
- retail price tier, In-App Purchase 28
- review notes, In-App Purchase 26
- rights, In-App Purchase 28

S

- saving In-App Purchase versions [42](#)
- screenshot for review, definition [44](#)
- screenshots
 - import [41](#)
 - In-App Purchase [26](#)
- searching [26](#), [28](#), [30](#), [32](#)
- servers [6](#)
- SKU number, definition [44](#)
- sort, columns [42](#)
- specifications [9](#)
- status
 - package [40](#)
 - uploading [37](#)
- support [5](#)
- supported formats [9](#)
- system requirements [6](#), [40](#)

T

- type
 - consumable [26](#)
 - non-consumable [26](#)
 - non-renewing subscription [26](#)

U

- uploading, app binary [36](#), [38](#)

W

- wholesale price tier default, In-App Purchase [29](#)



Apple Inc.
© 2015 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.

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.

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Apple, the Apple logo, Cocoa, Finder, iPad, iTunes, Mac, Mac OS, Macintosh, OS X, Pages, and Xcode are trademarks of Apple Inc., registered in the U.S. and other countries.

iTunes Store is a service mark of Apple Inc., registered in the U.S. and other countries.

App Store is a service mark of Apple Inc.

Intel and Intel Core are registered trademarks of Intel Corporation or its subsidiaries 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.

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.