ENDICIA LABEL SERVER™

Production Update January 27, 2013
Document Update January 23, 2013

Version 7.0

Copyright © 2004–2013 DYMO Endicia, A Newell Rubbermaid Company
All rights reserved.
This document is the property of DYMO Endicia, A Newell Rubbermaid Company and may not be reproduced, published, transmitted to other parties, in whole or in part, without written permission.

DYMO Endicia, A Newell Rubbermaid Company
385 Sherman Avenue
Palo Alto CA 94306

Contents

1. Overview **6**
About Endicia Label Server 6
About Representing Integrators, Partners, and Developers 6
ELS Path to Production and Checklist 7
Step 2 Details 8
Step 3 Details 9
Step 5 Details 10
Step 7 Details 11
Step 8 Details 13
Current Changes 14
2. The Endicia Label Server Web Service **20**
Web Methods 20
Proxy Server 21
Access the Endicia Label Server Web Service through SSL/HTTPS 21
Using the Endicia Label Server Web Service for the First Time 22
3. Requesting a Shipping Label API (Required) **23**
GetPostageLabel 23
LabelRequest XML Syntax 23
Notes 43
XML Request Code Samples 48
LabelRequestResponse XML Syntax 49
Notes 52
XML Response Code Samples 52
4. Buying Postage API (Required) **54**
RecreditRequest 54
RecreditRequest XML Syntax 54
RecreditRequestResponse XML Syntax 56
Postage Purchases 57
Endicia Bank-to-Bank Postage Payment Options 57
5. Changing the Pass Phrase API (Required) **59**
ChangePassPhraseRequest 59
ChangePassPhraseRequest XML Syntax 59
ChangePassPhraseRequestResponse XML Syntax 61
6. Resetting Suspsended Account Status API (Optional) **62**
Part 1: GetChallengeQuestion 62
GetChallengeQuestion XML Syntax 62
ChallengeQuestionResponse XML Syntax 63

Part 2: ResetSuspendedAccount 64
ResetSuspendedAccountRequest XML Syntax 64
ResetSuspendedAccountRequestResponse XML Syntax 66
7. Introduction to Calculating Postage and Fees 67
8. Calculate Postage and Fees API (Single) 68
PostageRateRequest 68
PostageRateRequest XML Syntax 68
PostageRateResponse XML Syntax 75
9. Calculate Postage And Fees API (Multiple) 78
PostageRatesRequest 78
PostageRatesRequest XML Syntax 78
PostageRatesResponse XML Syntax 83
10. Getting Account Status API 87
AccountStatusRequest 87
AccountStatusRequest XML Syntax 87
AccountStatusResponse XML Syntax 88
11. Creating and Updating Shared PES User Contact Information API90
SubmitUserContactInfo 90
SubmitUserContactInfoRequest XML Syntax 91
XML Request Code Sample 92
SubmitUserContactInfoResponse XML Syntax 92
XML Response Code Sample 93
12. Requesting a Refund API 94
RefundRequest 94
RefundRequest XML Syntax 94
RefundResponse XML Syntax 95
Request Refunds Through Your Endicia.com Account 98
View the Status of Previously Requested Refunds 98
Use the URL Method to Request a Refund 98
13. Requesting Package Status API 99
StatusRequest 99
StatusRequest XML Syntax 99
StatusResponse XML Syntax 101
Obtain Package Status Requests Through Your Endicia.com Account 104
14. Requesting Transaction Lists API 105
GetTransactionsListing 105
GetTransactionsListing XML Syntax 105
TransactionsListingResponse XML Syntax 108
View Transactions Listing Through Your Endicia.com Account 111

15. Using the SCAN Form API **112**
SCANRequest 112
SCANRequest XML Syntax 112
SCANResponse XML Syntax 114
Create a New SCAN Form Through Your Endicia Account 115
Create a SCAN Form History Report Through Your Endicia Account 116
16. Requesting a Carrier Pickup API's **117**
17. Scheduling a Carrier Pickup API **118**
CarrierPickupRequest 118
Notes 119
CarrierPickupResponse XML Syntax 120
18. Canceling a Carrier Pickup Request API **123**
CarrierPickupCancel 123
CarrierPickupCancelResponse XML Syntax 124
19. Changing a Carrier Pickup Request API **127**
CarrierPickupChange 127
CarrierPickupChangeResponse XML Syntax 129
20. Scheduling a Multiple Location Carrier Pickup Request API **132**
MultiLocationCarrierPickup 132
MultiLocationCarrierPickupResponse XML Syntax 132
21. Signing Up for an Account API **136**
UserSignupRequest 136
UserSignupRequest XML Syntax 136
UserSignupResponse XML Syntax 140
Additional User Signup Options 144
22. Additional Endicia API's **146**
Dial-A-ZIP XML 146
Endicia Printable Postage XML 146
Endicia XML 146
Integrated Partners 146
23. Printing Using EPL2 and ZPLII **147**
24. Customs Form Options **151**
What Qualifies as Domestic Mail 152
Populating the Customs Information 153
25. Frequently Asked Questions **155**
26. LabelType, LabelSize, and MailpieceShape by MailClass **158**
Appendix A. Change History **161**
Appendix B. Error Codes **189**

Appendix C. More Source Examples **196**
Coding Principles 196
Caution Statements 196
Coldfusion Examples 198
C# Examples 200
Perl Examples 205
PHP Examples 207
Ruby Examples 232
VisualBasic.NET Examples 234
Appendix D. Country Code Table **236**
Glossary **245**
Index **247**

1. OVERVIEW

About Endicia Label Server

Endicia Label Server (ELS) is an XML Web Service that enables you to create shipping labels with live prepaid postage *on the label!* You can:

- Print both domestic labels and international customs forms.
- Ship to APO/FPO destinations and international addresses.
- Qualify for postage discounts available only to PC Postage customers.

ELS can be integrated into any number of applications: from online shopping carts to Web sites; from fulfillment or distribution centers to multi-location, multi-user, and multi-carrier applications; and from consumer software applications to enterprise-wide solutions.

ELS supports expedited mail classes methods, such as Express Mail, Priority Mail, Critical Mail, and First-Class Package Services, as well as Standard Post, Library Mail, and Media Mail. ELS also supports all available USPS pricing as well as your NSA and GEPS contract pricing.

The ELS API uses a variety of Web methods, which allow you to seamlessly add USPS shipping into your applications without the need to change existing workflows, and enables you to manage your account, obtain package delivery status, produce SCAN forms, and schedule pickups with the USPS.

To enjoy the full benefits of ELS you send a label request to our Web Service in either HTTP POST or SOAP format. The ELS Web Service authenticates your Account ID and Pass Phrase and processes the label request. In as little as one second a response is returned containing the label with all your requested parameters.

About Representing Integrators, Partners, and Developers

You may represent one or more of these roles as you work with ELS:

- An *Integrator* uses ELS to print USPS labels for your fulfillment or distribution operations.
- A *Partner* uses ELS to integrate USPS labels into an end-user application, which you can offer to your customers.
- A *Developer*, combined with his or her programming skills, uses the ELS API to build a solution.

As an Integrator or Partner, you will work closely with developers and the Endicia Business Development team to produce an ELS solution to meet your needs.

ELS Path to Production and Checklist

Use this guide to speed your deployment of ELS. You can find details for each step on the page numbers indicated in the right column.

✓	Step	Page
	Agreement	
❑	1. With your MNDA (Mutual Non-Disclosure Agreement) completed, you can now move forward with your development. Keep the name, email address, and phone number of your Endicia Sales or Business Development representative available. You will be working very closely together.	
	Development	
❑	2. You can now begin building your interface using the testing server. At a minimum, you must create an interface for: • Printing labels (Label Request) • Setting and changing your Pass Phrase (Change Pass Phrase Request) • Purchasing postage (Recredit Request) The preceding items represent the core functions of ELS. We encourage you to integrate the other Web methods to meet your needs and provide a more robust solution to your users, such as refunding postage, obtaining package status, and consolidating your parcels wih a SCAN Request. There are several additional Web methods beyond the three core functions that you can use to enhance your application in ELS.	8
❑	3. After you have built and tested your interface, strengthen your application with strong security measures, especially if you will be granting access to others through a Web browser.	9
❑	4. Thoroughly test your finished application before proceeding to production.	
❑	5. To qualify for access to the production servers, you need to ensure you have completed all of the preceding steps and provided the following to your Endicia Business Development representative: • Complete the USPS Partner Profile (Exhibit B – Partner Description, in the ELS Agreement). • Review and sign your agreement. • Print sample labels generated in the format you will be using in production.	10
	Production	
❑	6. After all the preceding steps have been completed, you will be given the URL to the production servers, which will be used for all live postage transactions. a. Create your first account using the Production URL with your unique Partner ID or the Signing Up for an Account API. Be sure to make a note of the Internet Password and temporary PassPhrase that you created and use a valid email address—preferably the email address at which you will want to be contacted for	

	updates and notifications. b. Perform a PassPhrase Reset using the Changing the Pass Phrase API (Required) to set your permanent PassPhrase. c. Make your initial postage purchase using the Buying Postage API (Required). New accounts with Endicia allow a maximum purchase amount of $500 and a maximum balance of $500. Contact your Endicia Business Development representative to increase this limit.	
❑	7. Remember all the other Web methods included in the API that you can integrate to enhance your solution: You can also use our separate Dial-A-ZIP API to validate addresses before sending them to ELS. For more information, see the Additional Endicia API's section.	11
❑	8. Need technical support? Send email to LabelServer@endicia.com Monday through Friday 8:00 AM – 5:00 PM PST. If your question involves a specific Web method, always include the XML Request and Response in your email. If needed, a phone call can be scheduled with an engineering support specialist.	

Step 2 Details

Web Service System Requirements

The ELS Web Service requires that you have:

- A deployment computer with an Internet connection.
- The ability to send requests using HTTP 1.1 or SOAP 1.1 protocols.
- The ability to parse XML.

Required Web Methods

- **LabelRequest**. The LabelRequest Web method is the main and most complex Web method that you will implement. Most of your coding will be for the purpose of creating a user interface, which makes the process easy and automates as much of the decision-making process as possible. For the full specification, see Requesting a Shipping Label API (Required).
- **ChangePassPhrase**. For security purposes, you are required to reset the Pass Phrase after you create a new ELS account. Your organization or client may have a policy to change the Pass Phrase periodically in keeping with security protocols, so you should make this process easy, while also restricting access to authorized users. You cannot reuse a previously used Pass Phrase until you have cycled through at least four different Pass Phrases. For more information, see Changing the Pass Phrase API (Required).
- **RecreditRequest**. This is the means to make your initial purchase of prepaid postage. As with the ChangePassPhrase Web method, you may choose to restrict access to this functionality in your application. You may also configure an internal automatic purchase that would execute this Web method to purchase a predetermined amount when the account falls below a specified balance. For more information, see Buying Postage API (Required).

A listing of all Web methods is in Chapter 2. We encourage you to examine each one closely so that you may get the most from your ELS solution.

Test Server and WSDL

The ELS Web Service:

- Is a standard XML-based Web Service accessed through Web Services Description Language (WSDL).
- Can be programmatically accessed in any development environment that supports calling an XML Web Service using SOAP or HTTP POST.
- Has been tested extensively and documented for environments using Microsoft Visual Studio .NET tools for Web Service development as well as other environments.

The ELS Web Service Test Server is at:

https://www.envmgr.com/LabelService/EwsLabelService.asmx

The WSDL (Web Service Description Language) is at:

https://www.envmgr.com/LabelService/EwsLabelService.asmx?WSDL

The Test Server does not authenticate user credentials; therefore any characters that comply with the Size or Values restrictions can be used. The remaining chapters in this documentation describe all the Web methods in detail, and special chapters provide insights on customs forms, label request options, and much more.

Postage is MONEY

As you begin development, be aware that postage is essentially "money". As such, you need to make absolutely certain that your application does not allow postage to be printed accidentally. For example:

- Allowing the user to click a button multiple times before a transaction is completed will result in the unexpected purchase of multiple identical labels. Be sure to disable the button until each transaction is completed.
- Be certain that the label is successfully printed before deleting the image file or the Base64 data.

US postage is charged when the label is printed, not when it is scanned. Any time you have live postage on unused labels or a label that failed to print, you need to submit a Refund Request as soon as possible. You can use the Requesting a Refund API or request a refund through your account at www.endicia.com. It takes approximately two weeks for the USPS to authorize a credit to your account.

The USPS does not offer payment terms for postage. All postage must be paid at the time the postage label is printed through your prepaid account. You must safeguard your label production process to ensure that postage is not accidentally charged against your account and that your process restricts access to the API only to your authorized users.

Step 3 Details

ELS Security

It is absolutely necessary that the ELS solution you develop will:

- Safeguard the postage through a required Account ID and Pass Phrase for each transaction.
- Allow us to provide ELS functionality to Partners without requiring that the partners complete a security certification.

- Support labels that have a tracking barcode. This means that only mail classes that the USPS scans at some point are supported. These include:
    - *Delivery Confirmation*, *Signature Confirmation, or Certified Mail*: for First-Class parcels, Standard Post, Library Mail, Media Mail, Priority Mail, and Critical Mail.
    - *Express Mail*.
    - *Customs Numbers*: for First-Class Mail International, Priority Mail International, and Express Mail International.
    - Intelligent Mail barcode (IMb) and *Destination Confirm Service*: for First-Class letters and flats.

  There is an exception for "Mail Class Only", in which a barcode is produced on the label, but the USPS does not offer tracking information. You can usually still get limited tracking information through your endicia.com account or the Requesting Package Status API.

Developer Security

Use care to ensure that account numbers, user names, Internet passwords, and PassPhrases are kept secure.

Step 5 Details

Sample Label Requirements:

If you will be printing labels as an image (PNG, JPEG, or GIF), ensure that the dimensions have not been altered (based on the Label Size requested) or the aspect ratio has not been changed. If you are printing to EPL2- or ZPLII compatible printers, you must print sample labels to the printers that you will be using in production, scan or photograph them, and send them to your Endicia Business Development representative for approval.

The preceding sample is in EPL2 format (image has been reduced for space). Verify the following on your labels:

- Labels from thermal printers must print in solid black with no faded areas or grey "hollow" portions.
- The barcode must be centered across the bottom.
- The IBI (Information Based Indicia) in the upper-right must print clearly.
- No portion of the label may bleed off of the edge so that any portion of the image is cut off.

Step 7 Details

About Signing Up Accounts

One of the first things that a Partner must decide is how to set up accounts. Depending on how you are using ELS, each approach will differ. For more information and options, see Signing Up for an Account API.

If you will be providing our labels to your customers who are not also signing up for an Endicia account under your Partner account, the USPS requires that they be identified as a Shared PES (Postal Evidencing System) customer. Use the SubmitUserContactInfo Web method to submit this information to us the first time the customer uses your system for this purpose. ELS will return a unique identification for that customer, which you must include in all Label Request transactions in the <PartnerCustomerID> element.

Payment Considerations

One fundamental design constraint that impacts how many accounts you may set up is how the postage is paid. Currently, Endicia requires that an account can only have one financial way to fund postage. However, a Partner can have several accounts, each funded by a different source. Examples of different sources include, bank account, ACH pull or direct debit, ACH push or credit, check, wire transfer, and credit card. For more information about postage payment methods, see http://www.endicia.com/SignUp/Notice/PostagePaymentOptions/.

Bill Me Later® is available for customers on our Web site. However, the ELS API does not directly support its use. Customers may sign up with a supported payment method (credit card or ACH), then return to their account page at endicia.com and change the payment method to Bill Me Later.

CAPS (Centralized Account Processing System) and OMAS (Official Mail Accounting System) are also available options for qualified customers and agencies. Contact your Endicia Business Development representative to discuss participation in these programs.

There is no limit to the number of accounts that a Partner may open for themselves or on behalf of their customers (in the case of some of the Endicia Enterprise Solution customers).

Several different types of Partners use ELS, and depending on their business models, each Partner creates accounts in different ways. A few examples follow:

- A Partner could be an individual user or company wanting to implement a full Web Service solution into their application. In this case, the Partner can set up one Endicia account and allow all transactions to be processed through their application.

- A Partner could be an eCommerce Web site that has buyers and sellers. The Partner wants to allow the sellers to print shipping labels with postage. In this case, the Partner can set up one Endicia account and allow all transactions to flow through that account.

- A Partner might have a software product that is sold to other companies. Suppose that this is a large Web-based CRM Enterprise application. In this case, the Partner would not want to have the postage liability and instead would assign at least one Endicia account for each one of their customers.

- A Partner could be a CRM Enterprise customer (such as a Fortune 500 company) that has several funding mechanisms, and then they should have at least one Endicia account per funding mechanism.

- A Partner could be a third-party software provider for multi-carrier shipping systems seeking to integrate PC Postage into their application. In this case, the Partner could use either single or multiple accounts depending on their preference.

To explore these various options, contact your Endicia Business Development representative.

Endicia has created automated interfaces to allow creation of accounts through the API. Currently, Partners are provided with a URL to create accounts. For more information, see Signing Up for an Account API.

About Pass Phrases

You must complete the following steps to correctly obtain a production-ready Pass Phrase:

1. During the account sign-up stage, you must enter a temporary Pass Phrase.
2. After you receive the email message that contains the assigned account number, you must then use the ChangePassPhrase (or ChangePassPhraseXML) Web method to create a new Pass Phrase. *This is one of the important elements of the security model as required by the USPS*. If you are an approved Endicia Enterprise Partner, each customer will have their own Endicia account. You must include this functionality into your application so that your customers can set up accounts and then reset their Pass Phrase. This also allows them to independently reset and change their Pass Phrases.
3. Special note for Partners: When a new customer signs up for an account under your Partner ID, be sure the new user is led through the following steps so that their account can be activated without delay:
   a. Registration using the UserSignup Web method or the URL with your Partner ID.
   b. Change Pass Phrase from the temporary value they selected during signup, to a permanent Pass Phrase using the ChangePassPhraseRequest Web method. You can programmatically assign a temporary Pass Phrase within your process and then set a permanent Pass Phrase automatically to conceal this part of the process from the customer. It will make the process much easier for the customer so that the customer will be ready to print postage immediately.
   c. Prompt the customer to make their first postage purchase using the RecreditRequest Web method.

Do not reuse your old Pass Phrases.

About Internet Passwords

Endicia also carries the additional concept of the *Internet Password*. You use your Internet Password to gain access to the endicia.com Web site, where you can:

- View account transactions.
- Change account information.
- Change payment methods.
- Request refunds on unused labels.

The Internet Password does *not* enable you to buy and print postage.

About Printing Labels

When you create an account with Endicia, all account information and postage information is transferred to the USPS. Any postage that is purchased through ELS is transacted through Endicia, but the money never resides at Endicia. Instead, Endicia issues the transactions to take the money from the user's account to the USPS.

Refund Policy

Because you are printing shipping labels that have live postage, you must have a mechanism to recover costs if a misprint occurs.

The URL at http://www.endicia.com/ELS/refundpolicy.cfm describes the Endicia Refund Policy. The policy is current with the USPS rules and regulations. You can either embed the policy in a frame and display the policy to your customers or include the policy in your offering.

Because ELS returns an image of a label, the Partner should build in safeguards to allow the end user to make sure that they have successfully printed the label.

Note that because all the labels have some form of tracking, the refunds must be done electronically. Refunds can be requested through an API call or by logging in to the Endicia account page on www.endicia.com. All refunds must be submitted electronically. All electronic refunds must be requested within 10 days of printing the postage. Our system monitors the USPS Package Tracking System (PTS), and if the label does not appear within 14 days, then the amount of postage is refunded to the account.

Step 8 Details

ELS technical support is available Monday through Friday from 8:00 AM until 5:00 PM PST. We are closed on Federal holidays.

For technical issues, such as questions about API requests and responses, support is available by email at LabelServer@endicia.com.

For best results, always include the following information, as applicable, in your email:

- Your ELS account number
- Your name, phone number, and email address
- XML requests and responses as attachments or copy and paste the exact error message that you are receiving, and which API method is generating the message

We strive to respond to all email questions within an hour, but sometimes more time may be required to investigate your question and perform testing. If necessary, we may call you. For business-related issues, contact your Endicia Business Development representative.

Current Changes

Change # 1 Parcel Post renamed to Standard Post

Web Methods Affected:

- Requesting a Shipping Label API
- Calculate Postage and Fees API (Single and Multiple)

Description of Change:

Effective January, 20, 2013 the Parcel Post mail service is being renamed to Standard Post. This newly renamed service will be available at retail ONLY and includes electronic tracking at no additional cost. While this service will continue to be available to USPS Approved Shippers, commercial mailers should use the Parcel Select, barcoded, non-presorted mailing option in place of Standard Post (previously called Parcel Post).

Parcel Select is a comparable service to Parcel Post and is an economical ground-delivery option.

- Maximum weight is 70 pounds
- Extra Services: Parcel Select may use Adult Signature services, COD, Delivery Confirmation, insurance, return receipt for merchandise, and Signature Confirmation.
    - Delivery Confirmation is free.
- Parcel Select also has a mailing sub category called "Parcel Select LightWeight" for packages weighing less than one pound. More information is available at www.usps.com.

For USPS Approved Shippers:

➔ To request a Standard Post label, specify<MailClass> = 'StandardPost' (new value) and <MailPieceShape> = 'Parcel' or Null

Note: To minimize impact to our customers, the existing <MailClass> value 'ParcelPost' will be retained until 2014 Price Change. When this value is used, a Standard Post label will be returned.

SAMPLE LABEL **SAMPLE LABELREQUEST XML**

```
<LabelRequest LabelType="Default" LabelSize="4x6" ImageFormat="PNG">
<MailpieceShape>Parcel</MailpieceShape>
<MailClass>StandardPost<MailClass>
                :
                :
</LabelRequest >
```

For commercial mailers:

➔ To request a Parcel Select non-presorted, barcoded label, specify<MailClass> = 'ParcelSelect' <MailPieceShape> = 'Parcel' or Null, <SortType>='Nonpresorted' and <EntryFacility>='Other'

SAMPLE LABEL **SAMPLE LABELREQUEST XML**

```
<LabelRequest LabelType="Default" LabelSize="4x6" ImageFormat="PNG">
<MailpieceShape>Parcel</MailpieceShape>
<MailClass>ParcelSelect</MailClass>
                :
                :
<SortType>Nonpresorted</SortType>
<EntryFacility>Other</EntryFacility>
                :
</LabelRequest >
```

Backward Compatibility Note: Mandatory Change

Commercial mailers currently using Parcel Post must make sure that this service is replaced with Parcel Select service prior to the 2013 USPS price change, in order to avoid service disruptions.

However, in order to minimize customer impact during Price Change, when commercial mailers request Parcel Post service, a Parcel Select non-presorted label will be returned. This behavior will be deprecated in January 2014.

Change # 2 First-Class Mail International Parcels renamed to First-Class Package International Service (FCPIS)

Web Methods Affected:

- Requesting a Shipping Label API
- Calculate Postage and Fees API (Single and Multiple)

Description of Change:

Effective January 20, 2013 the First-Class Mail International Parcel service is being renamed to First-Class Package International Service (FCPIS). First-Class Mail International Letters, Postcards, and Flats services will remain unchanged.

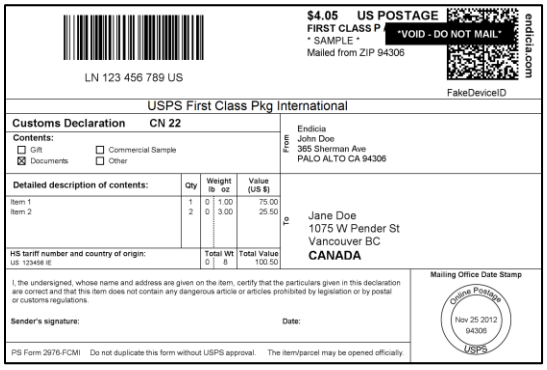
- Available pricing tiers: Commercial Plus, Commercial Base, and Retail. Default pricing for commercial mailers is Commercial Base.

- Same mailing requirements that applied to First-Class Mail International packages apply to First-Class Package International Service.

➔ To request a First-Class Package International Service label, specify<MailClass> = 'FirstClassPackageInternationalService' (new value) and <MailPieceShape> = 'Parcel' or Null

Note: In order to minimize impact to our customers, the existing <MailClass> value 'FirstClassMailInternational' will be retained until 2014 Price Change. When this value is used along with <MailPieceShape> = 'Parcel' or Null, a First-Class Package International Service (FCPIS) label will be returned.

SAMPLE LABEL



SAMPLE LABELREQUEST XML

```
<LabelRequest LabelType="Default" LabelSize="4x6"
LabelSubType="Integrated" LabelType="Interational"
ImageFormat="PNG">
<MailpieceShape>Parcel</MailpieceShape>
<MailClass>FirstClassPackageInternationalService</MailClass>
                :
                :
</LabelRequest >
```

Change # 3 Electronic USPS Delivery Confirmation International service

Web Methods Affected:

- Requesting a Shipping Label API
- Calculate Postage and Fees API (Single and Multiple)

Description of Change:

Effective January 20, 2013 the Electronic Delivery Confirmation service will be available for some international packages. This service provides mailers with the date and time that their package was delivered. If a delivery was attempted but not successful, this service provides mailers with the date and time of the delivery attempt.

- This is a free service.
- Service will be available for the following mail categories: First-Class Package International Service (formerly First-Class Mail International) parcels, Priority Mail International Flat Rate Envelopes, and Small Flat Rate Boxes.
- Service will be available in January 2013 for items shipped to Canada. Additional countries will be added at future dates.
- Labels will have markings to indicate that Electronic Delivery Confirmation Service is included.
- Delivery status on packages will be available on usps.com using the "Track and Confirm" feature. Additionally, tracking will also be available on www.dymoendicia.com/auth and by using the "Requesting Package Status" API.

➔ To request Electronic Delivery Confirmation Service for First-Class Package International Parcels, Priority Mail International Flat Rate Envelopes, and Small Flat Rate Boxes, specify the existing attribute DeliveryConfirmation="ON" in the Services element. If this attribute is completely omitted or is set to Null, this service will NOT be enabled.

➔ If the Electronic Delivery Confirmation Service has been requested, but is NOT available for the destination country specified in the LabelRequest, Label Server will return an error message.

SAMPLE LABEL

SAMPLE LABELREQUEST XML

```
<LabelRequest LabelType="Default" LabelSize="4x6"
LabelSubType="Integrated" LabelType="Interational"
ImageFormat="PNG">
<MailpieceShape>Parcel</MailpieceShape>
<MailClass>FirstClassPackageInternationalService</MailClass>
                :
<Services DeliveryConfirmation="ON"/>
                :
</LabelRequest>
```

Change # 4 Informational Updates

Web Methods Affected:

- Requesting a Shipping Label API
- Calculate Postage and Fees API (Single and Multiple)

Description of Change:

The following changes are for informational purposes only and updates will be taken care of in Endicia Label Server in addition to the required adjustments to USPS postage rates. Customers will not need to make any adjustments to their integrations:

#	Description
a	The weight limit for International Flat Rate Envelopes will be changed from 20 pounds to 4 pounds.
b	The previous limit of $100 guaranteed Express Mail International Insurance will be increased to $200. The price for the new, increased amount remains unchanged.
c	Priority Mail International Flat Rate shipments to Mexico will no longer receive the same rate as previous mailings to Canada and Mexico. Shipments to Mexico will now be included with the "All Other Countries" price tier. All other Priority Mail International parcels to Mexico remain unchanged.
d	Critical Mail will be available with a new signature option. This signature service will be added to the Critical Mail service at a more attractive rate than adding Signature Confirmation as an extra service. When Signature Confirmation is added to Critical Mail, Signature Confirmation will be automatically replaced by the new Critical Mail with Signature Service option.
e	The label layouts of the Integrated Customs Forms PS-2976 and PS-2976A have been modified per USPS guidelines.
f	Added new <imageformat> values: “PNGMONOCHROME”, “JPEGMONOCHROME”, “GIFMONOCHROME”, and “BMPMONOCHROME”: The monochrome image format values can be used to specifically request monochrome labels that will deliver up to 85% reduction in file size for image (BMP, JPEG, GIF and PNG) file formats.
g	Added the ability to suppress the fourth copy of the 4-part 2976A international customs form.
h	Added two new values added for <ContentsType> element: • HumanitarianDonations • DangerousGoods

2. THE ENDICIA LABEL SERVER WEB SERVICE

Web Methods

The following table lists all the Web methods of the Endicia Label Server Web Service:

Web Method Name	Description	Page
The following Web methods represent the core functions of Label Server and are required. The URL for testing these Web methods is https://www.envmgr.com/LabelService/EwsLabelService.asmx. The Endicia Label Server Web Service Web methods can be invoked using one of two protocols, HTTP POST or SOAP (over HTTP).		
GetPostageLabel or GetPostageLabelXML	Request a shipping label.	23
BuyPostage or BuyPostageXML	Add postage to an account.	54
ChangePassPhrase or ChangePassPhraseXML	Change Pass Phrase for an account.	59
The remaining Web methods are optional, and may be used to enhance your implementation.		
ChallengeQuestionRequest or ChallengeQuestionRequestXML ResetSuspendedAccountRequest or ResetSuspendedAccountRequestXML	Reset the status of an account that has been suspended due to multiple failed attempts to enter the correct Pass Phrase.	62
CalculatePostageRate or CalculatePostageRateXML	Calculate the postage and fees for a package.	68
CalculatePostageRates or CalculatePostageRatesXML	Calculate "all" domestic or international postage rates.	78
GetAccountStatus or GetAccountStatusXML	Request the status of an account.	87
The URL for testing the following Web method is: https://www.envmgr.com/LabelService/sharedpesservice.asmx		
SubmitUserContactInfo	Send Shared PES end-user contact information	83
The following Web methods use a different URL, https://www.endicia.com/ELS/ELSServices.cfc?wsdl. There is no testing server for these Web methods. Set the <Test> element to "Y" for testing. You can use HTTP GET with these Web methods.		
RefundRequest *	Request a refund.	94
StatusRequest *	Request status of a package.	99
GetTransactionsListing *	Request a list of transactions .	105

Web Method Name	Description	Page
SCANRequest *	Using the SCAN Form.	112
CarrierPickupRequest	Schedule a Carrier Pickup.	117
CarrierPickupCancel	Canceling a Carrier Pickup Request	123
CarrierPickupChange	Changing a Carrier Pickup Request	127
MultiLocationCarrierPickup	Scheduling a Multiple Location Carrier Pickup Request	132
The UserSignup Web method is required for most partner accounts and may also be used for managing multiple locations and other relationships where more than one account would be needed.		
UserSignup	Sign up for an account.	136

* You can also access the RefundRequest, StatusRequest, GetTransactionListing and SCANRequest through your account page at www.endicia.com. You can find details for each at the conlusion of the associated chapters.

Proxy Server

If the Endicia Label Server Web Service is being called from behind a proxy server that enforces limits on SOAP packet size, you may experience issues with sending and receiving large SOAP packets. For example, a call to the Endicia Label Server Web Service to render a label with a high-resolution results in a large SOAP packet being sent. You may need to bypass the proxy, set the SOAP packet size limit very high, or limit the amount of data you are requesting.

Access the Endicia Label Server Web Service through SSL/HTTPS

Because the Endicia Label Server Web Service resides on an SSL-enabled server, it will present a digital certificate to the client for authentication. The client must *explicitly* trust the certificate in order for the connection to be set up. If on the other hand, the client chooses not to trust the certificate, the connection will fail.

For further discussion and resolution of this issue, including a suggested workaround in .Net, see Article #823177 on Microsoft's support Web site at http://support.microsoft.com/default.aspx?scid=kb;en-us;823177

HTTP POST

The following is a general format of an HTTP POST request and response. The **placeholders** shown in bold font need to be replaced with actual values.

```
POST /LabelService/EwsLabelService.asmx/Web method Name HTTP/1.1
Host: www.envmgr.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

Request content
```

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

Response content
```

SOAP (Simple Object Access Protocol)

You must install a code library that can call a SOAP-based Web Service. Development tools, such as Visual Studio .Net, are set up to support Web Services out of the box.

Using the Endicia Label Server Web Service for the First Time

Before requesting your first shipping label using the Production Server, you must perform the following steps:

1. Sign up for a production account with the URL that will be supplied to you when you show us a test label and have a signed contract.
2. Use the ChangePassPhrase (or ChangePassPhraseXML) Web method to do a Pass Phrase reset by changing the temporary Pass Phrase assigned to your account during the sign-up process.
3. Use the BuyPostage (or BuyPostageXML) Web method to add postage to your account.

Now you are ready to use the GetPostageLabel (or GetPostageLabelXML) Web method to create shipping labels!

The following sections describe the input to, and response from, the Endicia Label Server Web Service Web methods in greater detail.

3. REQUESTING A SHIPPING LABEL API (REQUIRED)

GetPostageLabel

To request a shipping label, use the GetPostageLabel or the GetPostageLabelXML Web method of the Endicia Label Server Web Service.

The following is a sample HTTP POST request for GetPostageLabelXML, which requires a LabelRequest XML string as input. The **placeholders** shown need to be replaced with actual values:

```
POST /LabelService/EwsLabelService.asmx/GetPostageLabelXML HTTP/1.1
Host: www.envmgr.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

labelRequestXML=<LabelRequest> ... </LabelRequest>
```

The response contains a LabelRequestResponse XML string:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<LabelRequestResponse> ... </LabelRequestResponse>
```

LabelRequest XML Syntax

The following table describes the XML tags for the LabelRequest.

Important notes about all elements:

- Omitting any value causes the Endicia Label Server Web Service to use the default value for that element.
- *Italic* items are optional.
- The Endicia Label Server Web Service requires a valid destination address to process a shipping label. For more information on address validation, see Additional Endicia API's.

Table 3-1: LabelRequest XML Elements

Element	Type	Size or Values	Description
LabelRequest	**Root**	**-**	**The root node for the XML request.**
Test	*Attribute*	*YES* *NO*	*Use sample postage for testing.* *Use live postage. (Default)*
LabelType	*Attribute*	*Default* *CertifiedMail* *DestinationConfirm* *Domestic* *International*	*Create label based on mail class. (Default)* *Create Certified Mail label (PS Form 3800 required).* *Create IMb label using Destination Confirm service. Create domestic label.* *Requires use of the LabelSubtype element. See Note 1.* *Create international label.* *When the value of this element is set to Domestic or International, the label will be returned as separate images within the Label node of the LabelRequestResponse XML.*
LabelSubtype	*Attribute*	*Integrated* *None*	*Create an integrated label. The integrated form type must be specified in the IntegratedFormType element.* *No label subtype (Default)* *Required when label type is Domestic or International.*

Element	Type	Size or Values	Description
LabelSize	*Attribute*		*For Default label type:*
		4x6	*4" W x 6" H (Default)*
		4x5	*4" W x 5" H*
		4x4.5	*4" W x 4.5" H*
		DocTab	*4" W x 6.75" H, Eltron Doc-Tab label*
		6x4	*6" W x 4" H*
		4x8	*4"W x 8" H (Prints referenceID, Costcenter, Tracking Number, RubberStamp1, RubberStamp2, RubberStamp3 field values at the bottom of the label)*
			For DestinationConfirm label type:
		7x3	*7" W x 3" H (Default)*
		6x4	*6" W x 4" H*
		Dymo30384	*DYMO #30384 2-part internet label (2.25" x 7.5")*
		EnvelopeSize10	*#10 Envelope*
		Mailer7x5	*7" W x 5" H*
			For CertifiedMail label type:
		4x6	*4" W x 6" H (Default)*
		7x4	*7" W x 4" H*
		8x3	*8" W x 3" H*
		Booklet	*9" W x 6 "H envelope*
		EnvelopeSize10	*#10 Envelope*
			For International label type:
		Null or any other value	*8.5" x 5.33" (Default)*
		4x6c	*6" W x 4" H*
ImageFormat	*Attribute*	*EPL2*	*EPL2 programming language*
		GIF	*GIF (24-bit color)*
		GIFMONOCHROME	*GIF (monochrome)*
		JPEG	*JPEG (24-bit color)*
		JPEGMONOCHROME	*JPEG (monochrome)*
		PDF	*PDF*
		PNG	*PNG (Default)*
		PNGMONOCHROME	*PNG(monochrome)*
		ZPLII	*ZPL II programming language*
		BMPMONOCHROME	*BMP (monochrome)*
			For more information, see Note 2.

Element	Type	Size or Values	Description
ImageResolution	*Attribute*	*203* *300*	*203 dpi* *300 dpi* *Image resolution defaults to 300 dpi for domestic labels as well as labels created using LabelType of International.* *Image resolution defaults to 203 for labels with ImageFormat of EPL2 or ZPLII.* *If this element value is supplied when ImageFormat is set to EPL2 or ZPLII, it must be set to 203.*
ImageRotation	*Attribute*	*None* *Rotate90* *Rotate180* *Rotate270*	*No rotation (Default).* *Rotate label image 90 degrees.* *Rotate label image 180 degrees.* *Rotate label image 270 degrees.* *For labels in EPL2 and ZPLII format, selecting Rotate180 prints from top of image buffer; selecting None prints from bottom of image buffer.* *Rotate90 and Rotate270 are not available for labels in EPL2 and ZPLII format.*
RequesterID	Text	50	Requester ID (also called Partner ID) uniquely identifies the system making the request. Endicia assigns this ID. The Test Server does not authenticate the RequesterID. Any text value of 1 to 50 characters is valid.
AccountID	Numeric	6	Account ID for the Endicia postage account. The Test Server does not authenticate the AccountID. Any 6-digit value is valid.

Element	Type	Size or Values	Description
PassPhrase	Text	64	Pass Phrase for the Endicia postage account. The Test Server does not authenticate the PassPhrase. Any text value of 1 to 64 characters is valid.

Element	Type	Size or Values	Description
MailClass	Text		Domestic:
		Express	Express Mail
		First	First-Class Package Service and First-Class Mail Parcel
		LibraryMail	Library Mail
		MediaMail	Media Mail
		StandardPost	Standard Post (formerly called Parcel Post)
		ParcelSelect	Parcel Select or Parcel Select Lightweight
		Priority	Priority Mail
		CriticalMail	Critical Mail
			International:
		ExpressMailInternational	Express Mail International
		FirstClassMailInternational	First-Class Mail International
		FirstClassPackageInternationalService	First-Class Package International Service
		PriorityMailInternational	Priority Mail International
		GXG	Global Express Guaranteed *(For future use)*
		For DHL Consolidator No-Postage labels:	
		DHLGMSMParcelPlusExpedited	DHL Global Mail Parcel Plus Expedited Service
		DHLGMSMParcelPlusStandard	DHLGM Parcel Plus Standard Service
		DHLGMSMBPMExpedited	DHLGM Bound Printed Matter Expedited Service
		DHLGMSMBPMGround	DHLGM Bound Printed Matter Ground Service
		DHLGMSMCatalogBPMExpedited	DHLGM Catalog Bound Printed Matter Expedited Service
		DHLGMSMCatalogBPMGround	DHLGM Catalog Bound Printed Matter Ground Service
		DHLGMSMMediaMailGround	DHLGM Media Mail Ground Service
		DHLGMSMParcelsExpedited	DHLGM Parcel Expedited Service
		DHLGMSMParcelsGround	DHLGM Parcel Ground Service
		DHLGMSMPriorityMail	DHLGM Priority Mail Service
		DHLGMSMFirstClassProduct	DHLGM First Class Product
		DHLGMSMFirstClassParcels	DHLGM First Class Parcel Service
		For DHL Consolidator No-Postage labels:	
		IPA	International Priority Airmail
		ISAL	International Surface Air Lift

Element	Type	Size or Values	Description
DateAdvance	*Numeric*	*0-7 (default range)*	*The number of days to advance date on the indicium.* *When used, this element advances the date on the indicium by the number of days selected from the range of values.* *Default range: (0-7) days.*
WeightOz	Numeric	4.1	Weight of the package, in ounces.
MailpieceShape	*Text*	*Card* *Letter* *Flat* *Parcel* *LargeParcel* *IrregularParcel* *FlatRateEnvelope* *FlatRateLegalEnvelope* *FlatRatePaddedEnvelope* *FlatRateGiftCardEnvelope* *FlatRateWindowEnvelope* *FlatRateCardboardEnvelope* *SmallFlatRateEnvelope* *SmallFlatRateBox* *MediumFlatRateBox* *LargeFlatRateBox* *DVDFlatRateBox* *LargeVideoFlatRateBox* *RegionalRateBoxA* *RegionalRateBoxB* *RegionalRateBoxC* *LargeFlatRateBoardGameBox* Containers for Open & Distribute services: *HalfTrayBox* *FullTrayBox* *EMMTrayBox* *FlatTubTrayBox*	*Shape of the mailpiece.* *Use LargeParcel instead of MailpieceDimensions to indicate Balloon Rate.* *IrregularParcel is used for Parcel Select LightWeight only.* *If MailpieceShape is set to LargeParcel, it overrides the value supplied in MailpieceDimensions.* *For more information, see the LabelType, LabelSize and MailpieceShape by MailClass.* *PMOD container-specific pricing available*
MailpieceDimensions	*Node*		*Dimensions of the mailpiece.* *For more information, see Note 3.*

Element	Type	Size or Values	Description
Length	Numeric	3.3	Length of the mailpiece. For best rates, always enter the longest dimension as the length.
Width	Numeric	3.3	Width of the mailpiece.
Height	Numeric	3.3	Height (or thickness) of the mailpiece.
PackageTypeIndicator	*Text*	*Null* *Softpack*	*Package Type is regular (Default).* *Commercial Plus Cubic price for the soft pack packaging alternative.*
Pricing	*Text*	*Retail*	*Retail pricing*
Machinable	*Text*	*TRUE* *FALSE*	*Mailpiece is machinable. (Default)* *Mailpiece is non-machinable.* *If a Parcel Select mailpiece marked as machinable is over 35 lbs. in weight, it will automatically be charged the non-machinable price.*
The following six elements are for Express Mail only.			
SignatureWaiver	*Text*	*TRUE* *FALSE*	*Request waiver of signature for receipt of mailpiece. (Default)* *Request signature for receipt of mailpiece.* **Note:** *As of release version 6.5, the default value for this element has been changed from "FALSE" to "TRUE". Endicia recommends that the value of this element be set to TRUE.*
NoWeekendDelivery	*Text*	*TRUE* *FALSE*	*Request that mailpiece should NOT be delivered on a Saturday.* *Request that mailpiece can be delivered on a Saturday (Default)*
ServiceLevel	*Text*	*NextDay2ndDayPOToAddressee*	*Next Day/Second Day Post Office to Addressee Service.*

Element	Type	Size or Values	Description
SundayHolidayDelivery	*Text*	*TRUE* *FALSE* *SUNDAY* *HOLIDAY*	*Request Sunday and Holiday Delivery Service.* *Do not deliver on Sunday or holiday. (Default)* *Request Sunday Delivery Service* *Request Holiday Delivery Service* *For more information, see Note 4.*
ShipDate	*Date*	*MM/DD/YYYY*	*Date mailpiece is shipped.* *Sunday/Holiday Delivery Service. Ignored for other mail classes.*
ShipTime	*Time*	*HH:MM AM or HH:MM PM*	*Time mailpiece is shipped.* *Sunday/Holiday Delivery Service. Ignored for other mail classes.*
The following three elements are for presorted services only			
AutomationRate	*Text*	*TRUE* *FALSE*	*Use applicable automation rate for selected mail class.* *Use retail price. (Default)* *Available only for letter shape mailpiece using First-Class.*
SortType	*Text*	*NDC* *FiveDigit* *MixedNDC* *Nonpresorted* *Presorted* *SCF* *SinglePiece* *ThreeDigit*	*Sort level for applicable mail classes.* *Required for Parcel Select and Parcel Select LightWeight.* *Defaults to SinglePiece for mailpieces that do not require a sort type.*

Element	Type	Size or Values	Description
EntryFacility	*Text*	*DNDC* *DDU* *DSCF* *ONDC* *Other*	*Postal facility where the mail is entered. If this element is not set to Other, the ZIP Code of this facility must be specified in POZipCode.* *Required for Parcel Select and Parcel Select LightWeight.* *Destination NDC* *Destination Delivery Unit* *Destination Sectional Center Facility* *Origin NDC* *Other postal facility (Default)*
POZipCode	*Text*	*5*	*ZIP Code of Post Office or collection box where item is mailed. May be different than FromPostalCode.* *Used for determining the zone and calculating the postage price.* *Required when EntryFacility is not set to Other. The value of this element must contain the ZIP Code of the postal facility specified in EntryFacility.*
IncludePostage	*Text*	*TRUE* *FALSE*	*Print postage on the label. (Default)* *Do not print postage on the label*
PrintConsolidatorLabel	*Text*	*TRUE* *FALSE*	*Print Consolidator "no-postage" Label* *Do not print Consolidator "no-postage" label (Default)* **Note:** *When the value is set to "TRUE", the IncludePostage element value must be set to "FALSE".*

Element	Type	Size or Values	Description
ReplyPostage	*Text*	*TRUE*	*Print reply postage on the label, which means the Sender's and Recipient's address will be swapped when printing.*
		FALSE	*Do not print reply postage. (Default)* *Can only be with label type of Default or DestinationConfirm.* *Not available for Express Mail, international mail, integrated labels, or when Return Receipt is requested.*
ShowReturnAddress	*Text*	*TRUE*	*Print sender's address on the label. (Default)*
		FALSE	*Do not print sender's address on the label.* *Defaults to TRUE for integrated labels.* *Even though this element can be used to hide the return address in the label, the USPS rules requires that a return address must appear on the mailpiece in specific circumstances. For more information, see the USPS Domestic Mail Manual.*
Stealth	*Text*	*TRUE*	*Turn on the use of "Stealth Postage" or hidden postage price. (Default)*
		FALSE	*Turn off Stealth Postage.* *For more information, see Note 5.*
ValidateAddress	*Text*	*TRUE*	*Validate all addresses. (Default)*
		FALSE	*Bypass address validation (requires partner to do Address Validation).*

Element	Type	Size or Values	Description
LiveAnimalSurcharge	*Text*		*Used only for shipping live animals, such as day-old chicks.*
		TRUE	*Surcharge is applied. Surcharge is based on weight and is only added if the destination zone is greater than 4.*
		FALSE	*Surcharge is not applied (Default).*
Services	*Node*		*Special Services requested for the package.*
See Note 6 on tracking methods.			
DeliveryConfirmation	*Attribute*	*OFF*	*Delivery Confirmation not requested.*
		ON	*Delivery Confirmation requested.*
SignatureConfirmation	*Attribute*	*OFF*	*Signature Confirmation not requested. (Default)*
		ON	*Signature Confirmation requested.*
CertifiedMail	*Attribute*	*OFF*	*Certified Mail not requested.*
		ON	*Certified Mail requested (PS Form 3800 required).*
RestrictedDelivery	*Attribute*	*OFF*	*Restricted Delivery not requested. (Default)*
		ON	*Restricted Delivery requested.*
ReturnReceipt	*Attribute*	*OFF*	*Return Receipt not requested. (Default)*
		ON	*Return Receipt requested (must affix a completed Return Receipt Form 3811 to the mailpiece and take it to the retail USPS counter).* *Can only be used with label type of Default or DestinationConfirm.* *Not supported for International Mail.*

Element	Type	Size or Values	Description
ElectronicReturnReceipt	*Attribute*	*OFF* *ON*	*Electronic Return Receipt not requested. (Default)* *Electronic Return Receipt requested.* *To receive Electronic Return Receipt delivery information, the partner must be registered as a Bulk Proof of Delivery Client with the USPS and develop the software to download the extract file from USPS. The Mailer ID assigned by the USPS for this purpose must be specified in the BpodClientDUNSNumber element.*
HoldForPickup	*Attribute*	*OFF* *ON*	*Hold For Pickup not requested (Default).* *Hold For Pickup requested For more information, see Note 7.*
OpenAndDistribute	*Attribute*	*OFF* *ON*	*Open and Distribute service not requested (Default)* *Open and Distribute service requested* *Open and Distribute Service is available only for MailClass of Priority (for PMOD) or Express (for EMOD).*
COD	*Attribute*	*OFF* *ON*	*COD not requested. (Default)* *COD requested.* *(Must affix a completed COD Form 3816 to the mailpiece and take it to the retail USPS counter).*

Element	Type	Size or Values	Description
InsuredMail	*Attribute*	*OFF* *UspsOnline* *Endicia*	*No insurance requested. (Default)* *USPS online insurance requested.* *Endicia Parcel Insurance requested (Maximum insurable value: $10,000)* *USPS online insurance is available only for mailpieces with Delivery Confirmation or Signature Confirmation.* *USPS insurance is not allowed for International Mail or when Stealth or ReplyPostage is set to TRUE.* *Endicia insurance fee is not included in the postage price. It is billed to your account.*
AdultSignature	*Attribute*	*OFF* *ON*	*Adult Signature not requested. (Default)* *Adult Signature requested.*
AdultSignatureRestrictedDelivery	*Attribute*	*OFF* *ON*	*Adult Signature Restricted Delivery not requested. (Default)* *Adult Signature Resticted Delivery requested.*
RegisteredMail	*Attribute*	*OFF* *ON*	*Registered Mail service is not requested.* *Registered Mail service is requested.* **Note:** *Mailer must affix an USPS issued Label200 to the mail piece, in addition to the Shipping Label*
CODAmount	*Currency*	*5.2*	*COD amount to collect.* *Required if COD is requested for the mailpiece.*
InsuredValue	*Currency*	*5.2*	*Insured value of the mailpiece.* *Required if insurance or COD is requested for the mailpiece.*
The following four elements are user-defined.			

Element	Type	Size or Values	Description
CostCenter	*Numeric*	*8*	*Cost-center code for accounting purposes.*
CostCenterAlphaNumeric	*Text*	*50*	*Cost-center code for accounting purposes.* **Note:** *This element serves the same purpose as the CostCenter element above. Use this element if the cost center values are alphanumeric. Do not use both these elements together.*
ReferenceID	*Text*	*50*	*A reference value for your logs.*
ReferenceID2	*Text*	*50*	*A reference value for your logs.*
ReferenceID3	*Text*	*50*	*A reference value for your logs.*
ReferenceID4	*Text*	*50*	*A reference value for your logs.*
PartnerCustomerID	Text	120	A unique identifier for the partner's end-user printing the label. Use this element for Shared PES accounts.
PartnerTransactionID	Text	25	A unique identifier for the partner's end-user's transaction such as invoice, transaction number, and so on.
BpodClientDunsNumber	*Numeric*	*9*	*Mailer ID of the partner assigned by USPS.* *Allows the partner to receive Electronic Return Receipt information from USPS in an extract file uniquely generated for them, provided they have established themselves with USPS as a Bulk Proof of Delivery Client.*
For details and limitations on using Rubber Stamps, see Note 8.			
RubberStamp1	*Text*	*50*	*User-supplied text to print on the label.*
RubberStamp2	*Text*	*50*	*User-supplied text to print on the label.*
RubberStamp3	*Text*	*50*	*User-supplied text to print on the label.*
ResponseOptions	*Node*		*Optional XML elements to include in the LabelRequestResponse.*
PostagePrice	*Attribute*	*TRUE* *FALSE*	*TRUE means the response contains the PostagePrice node.*

Element	Type	Size or Values	Description
The "From..." elements, excluding FromPhone and FromEmail, make up the Sender's information. See Note 9 for limitations.			
FromName	*Text*	*47*	*Sender's name* *Either FromName or FromCompany must contain a value.* *For customs forms, this element must contain at least two words.*
FromCompany	*Text*	*47*	*Sender's company name* *Either FromName or FromCompany must contain a value.*
ReturnAddress1	Text	47	First delivery address line of sender
ReturnAddress2	*Text*	*47*	*Second delivery address line of sender*
ReturnAddress3	*Text*	*47*	*Third delivery address line of sender.* *Do not use when label type is Domestic or International and a label subtype value is supplied.*
ReturnAddress4	*Text*	*47*	*Fourth delivery address line of sender.* *Do not use when label type is Domestic or International and a label subtype value is supplied.*
FromCity	Text	50	Sender's city Allowed characters: A-Z, a-z, hyphen, period, space
FromState	Text	25	Sender's state or province. If ValidateAddress is FALSE, this element must contain a valid 2-character state code.
FromPostalCode	Text	10	Sender's postal code. The format is either ZIP5 only or ZIP+4 for US addresses.
FromZIP4	*Text*	*4*	*+4 add-on for US addresses.* *Ignored if FromPostalCode contains the ZIP4 value.*
FromCountry	*Text*	*50*	*Sender's country. This value should be left blank for USA addresses.*

Element	Type	Size or Values	Description
FromPhone	*Text*	*10*	*Phone number of sender (required for Express Mail and international mail)* *10 digits required (including area code) with no punctuation. Use format: 2125551234*
FromEMail	*Text*	*64*	*Email address of sender.*
The "To..." elements, excluding ToPhone and ToEmail, make up the Addressee information. See Note 9 for limitations.			
ToName	*Text*	*47*	*Recipient's name* *For Express Mail and international mail: Either ToName or ToCompany must contain a value.*
ToCompany	*Text*	*47*	*Recipient's company name* *For Express Mail and international mail: Either ToName or ToCompany must contain a value.*
ToAddress1	Text	47	First delivery address line of recipient A value is optional only when ToCompany contains a value. Express Mail labels are limited to four lines in the destination address for all label sizes. 4 × 5 and 4 × 4.5 labels are limited to five lines in the destination address.
ToAddress2	*Text*	*47*	*Second delivery address line of recipient*
ToAddress3	*Text*	*47*	*Third delivery address line of recipient.* *Do not use with label type of Domestic.*
ToAddress4	*Text*	*47*	*Fourth delivery address line of recipient.* *Do not use with label type of Domestic.*
ToCity	Text	50	Recipient's city For Domestic Mail, allowed characters are: A-Z a-z hyphen period space

Element	Type	Size or Values	Description
ToState	Text	2 or 25	Recipient's state or province. If ValidateAddress is FALSE domestic mail, this element must contain a valid 2-character state code. Required for Domestic Mail, optional for International Mail.
ToPostalCode	Text	 5 15	Recipient's postal code. For Domestic Mail, the format is ZIP5 (required). For International Mail (optional).
ToZIP4	*Text*	*4*	*+4 add-on for US addresses.*
ToDeliveryPoint	*Text*	*2*	*+2 Delivery Point for US addresses.*
ToCountryCode	*Text*	*2*	*Two-character country code of the recipient's country.* *Refer to* Country Code Table*, for a table of valid country names.* *Required when label type is International and a label subtype value is supplied.*
ToPhone	*Text*	 *10* *30*	*Recipient's Phone Number.* *For domestic mail: 10 digits including area code with no punctuation. For example: 2125551234. If supplied value is not in the correct format, it will be ignored.* *For international mail, up to 30 digits with no punctuation.*
ToEMail	*Text*	*64*	*Email address of recipient.*
The following nine elements are for Hold for Pickup service. For more information, see Note 7.			
HfpFacilityID	*Text*	*50*	*Destination Facility ID.*
HfpFacilityName	*Text*	*50*	*Destination Facility Name.*
HfpFacilityAddress1	*Text*	*50*	*Destination Facility Address.*
HfpFacilityCity	*Text*	*50*	*Destination Facility City.*
HfpFacilityState	*Text*	*50*	*Destination Facility State.*
HfpFacilityPostalCode	*Text*	*25*	*Destination Facility 5-Digit ZIP Code.*
HfpFacilityZIP4	*Text*	*5*	*Destination Facility +4.*

Element	Type	Size or Values	Description
HfpEmailAddress	*Text*	*64*	*Email address of the addressee. An email address or SMS number is required if Service HoldForPickup is selected.*
HfpSMS	*Text*	*10*	*SMS phone number of the addressee. An email address or SMS number is required if Service HoldForPickup is selected.*
The following five elements are for Open and Distribute Service.			
OpenAndDistributeFacilityType	*Text*	*SCF* *DDU* *NDC* *ADC* *ASF* *NDC*	*Sectional Center Facility* *Destination Delivery Unit* *Network Distribution Center* *Area Distribution Center* *Auxiliary Service Facility* *Bulk Mail Center*
OpenAndDistributeFacilityName	*Text*	*50*	*Required when OpenAndDistributeFacilityType = 'DDU'*
OpenAndDistributeTray	*Text*	*True* *False*	*Enclosed mail must be placed in a tray if the total weight is less than 5 pounds*
OpenAndDistributeMailClassEnclosed	*Text*	*Standard Mail Letters* *Standard Mail Flats* *Standard Mail Parcels* *Standard Mail Mix* *Other*	*Mailpieces enclosed*
OpenAndDistributeMailClassOther	*Text*	*32*	*Required when OpenAndDistributeMailClassEnclosed = 'Other'*
CustomsInfo Node Customs Forms Elements (For more information, see Customs Form Options.)			
IntegratedFormType	*Text*	*Form2976* *Form2976A*	*Form 2976 (short form: use this form if the number of items is 5 or fewer.)* *Form 2976-A (long form: use this form if number of items is greater than 5.)* *Required when the label subtype is Integrated.*
CustomsCertify	*Text*	*TRUE* *FALSE*	*TRUE means the customs information is certified to be correct and the CustomsSigner name should be printed.*

Element	Type	Size or Values	Description
CustomsSigner	*Text*	*47*	*Name of person certifying that the customs information is correct. This name prints on the customs form in place of a signature if CustomsCertify is TRUE.* *Required if CustomsCertify is TRUE.*
CustomsSendersCopy	*Text*	*TRUE* *FALSE*	*Default* *Sender's copy is one of the 4-part international customs PS-2976A forms. This copy will print if value is set to TRUE.* ***Note:*** *This applies only to international 4x6 labels.*
CustomsInfo	*Node*		*Customs Declaration* *Required when using Integrated labels..*
ContentsType	*Text*	*Documents* *Gift* *Merchandise* *Other* *ReturnedGoods* *Sample* *HumanitarianDonation* *DangerousGoods*	*Category of the customs items.* *Value of ReturnedGoods is not available for labels using Form 2976.*
ContentsExplanation	*Text*	*25*	*Explanation of the customs items.* *Required if ContentsType is Other.*
RestrictionType	*Text*	*None* *Other* *Quarantine* *SanitaryPhytosanitaryInspection*	*Default*
RestrictionComments	*Text*	*25*	*Explanation of restrictions if RestrictionType is not None.*
SendersCustomsReference	*Text*	*14*	*Sender's customs reference.*
ImportersCustomsReference	*Text*	*40*	*Importers customs reference.*
LicenseNumber	*Text*	*16*	*License number.*
CertificateNumber	*Text*	*12*	*Certificate number.*
InvoiceNumber	*Text*	*15*	*Invoice number.*

Element	Type	Size or Values	Description
NonDeliveryOption	*Text*	*Abandon* *Return*	*Non-delivery instructions.* *Treat as abandoned.* *Return to sender. (Default)*
InsuredNumber	*Text*	*13*	*For future use.*
EelPfc	*Text*	*35*	*Exemption or Exclusion Legend (EEL) or a Proof of Filing Citation (PFC).Required for shipments to an international destination or to an overseas U.S. Territory.* *It is recommended to supply a value for this element as the USPS will likely require it in the near future.*
CustomsItems	Node		*Must contain at least 1 but no more than 999 CustomsItem nodes.*
CustomsItem		*[1 .. 999]*	
Description	Text	50	Description of the customs item.
Quantity	Numeric	3	Quantity of the customs item. Must be greater than zero.
Weight	Numeric	4.2	Weight of the customs item. Must be specified in whole ounces and greater than zero and cannot exceed 1120 ounces (70 pounds).
Value	Currency	5.2	Value of the customs item. Must be greater than zero.
HSTariffNumber	*Text*	*6*	*6-digit HS tariff number.*
CountryOfOrigin	*Text*	*2*	*Two character country code of the country where the customs items originated.* *Refer to the Country Code Table for a list of valid country codes.*
The following fields are for future use.			
GXGFedExTrackingNumber	*Text*	*12*	*For Future Use.*
GXGUSPSTrackingNumber	*Text*	*10*	*For Future Use.*

Notes

1. **Domestic LabelType.** The Domestic LabelType is used with the LabelSubtype attribute and CustomsInfo node for APO/FPO/DPO addresses and for U.S. Territories and Possessions. Limitations on available tracking methods apply. For more information, see Note 5.

2. **ImageFormats.** Note the following:

- Only GIF is available for international labels created using LabelType of Default (not recommended).
- The monochrome image format values can be used to specifically request monochrome labels that will deliver up to 85% reduction in file size for image (BMP, JPEG, GIF and PNG) file formats.
- EPL2 and ZPLII are supported for:
    - Default label type for domestic mail classes.
    - Domestic label type when used with integrated form type 2976.
    - International label type when used with integrated form type 2976A (4x6c label type) for PMI/EMI.
    - International label type when used with First-Class Mail International.
    - DestinationConfirm services for First-Class Flats, Letters, and Cards.
- The label image returned in the response for EPL2 and ZPLII labels contain binary data, which require special handling. For more information, see Printing Using EPL2 and ZPLII.

For more information, see LabelType, LabelSize, and MailpieceShape by MailClass table on page 158.

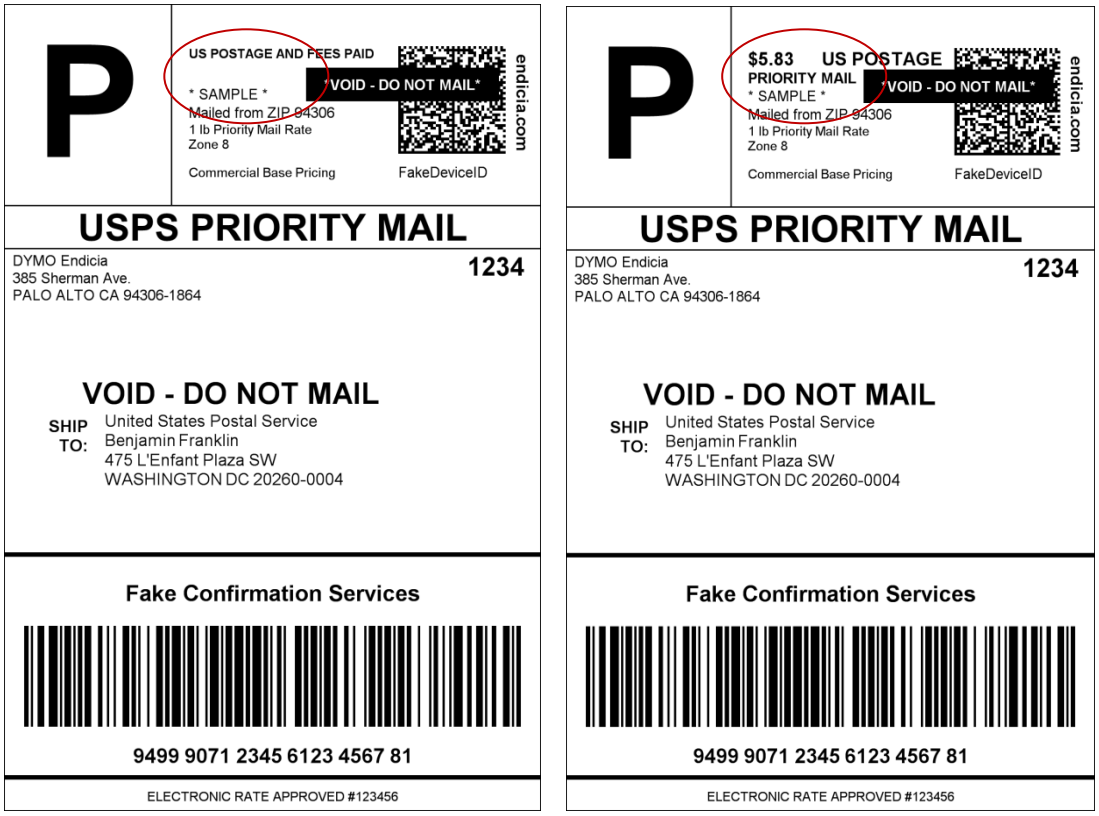
3. **Mailpiece Dimensions.** Required for calculating Commercial Plus Cubic pricing and for Priority Mail mailpieces going to Zones 5-8 which are over 1 cubic foot. All values must be in inches. See http://pe.usps.com/text/dmm300/123.htm for properly calculating balloon and dimensional weights. If you know that balloon or dimensional weights will apply, but do not want to enter the measurements, choose MailpieceShape LargeParcel.

4. **Express Mail Sunday and Holiday Service.** For an additional fee, your Express Mail mailpiece will be delivered on Sundays and Federal holidays. Express Mail Sunday and Holiday delivery service is not available to all destinations. When shipping a mailpiece with the MailClass of ExpressMail and SundayHolidayDelivery set to "Sunday" or "Holiday", the Endicia Label Server performs the following steps:
    1. Verify if the next day or second day after the ShipDate is on a Sunday or a Holiday. If it is not, the Endicia Label Server returns error code 11111.
    2. Retrieve the Express Mail Service Commitments from the USPS servers using the origin and destination ZIP Codes and ShipDate.
        - If no Service Commitments are received for Next Day 12 PM, Next Day 3PM, or 2-Day delivery service, the Endicia Label Server returns error code 11112.
        - If Service Commitments are received for two delivery days only, then the Endicia Label Server returns error code 11113.
        - In case of an error communicating with the USPS servers, the Endicia Label Server can return either error code 11102, ExpressMailAPIError, or error code 11110.
    3. Verify the day when the Service Commitment occurs. If it does not occur on a Sunday or a Holiday, the Endicia Label Server returns error code 11113.
    4. Cycle through all locations for the Service Commitments to determine if at least one location accepts Express Mail shipments past the supplied ShipTime. If all cut-off times have expired, the Endicia Label Server returns error code 11114. ShipTime is optional; if it is not supplied, it defaults to 12:01 AM, essentially accepting any location in this step.

       For a label or price request, the Endicia Label Server returns the appropriate response with the $12.50 Sunday-Holiday Delivery premium included in the postage. For a label

request, if the destination ZIP Code does not deliver Express Mail on Sundays or holidays, the Endicia Label Server returns error code 11115.

5. **Stealth Postage™ Labels.** The Stealth element permits you to effectively "hide" the postage on your postage label. The label on the left is a Stealth Postage label, which does not show the postage price. The label on the right shows the postage price.



Stealth cannot be used with COD, USPS Insurance, Registered Mail, Automation rate, LabelSize of EnvelopeSize10 and Card shape mailpieces.

6. **Tracking Methods.** Label Server requires a tracking method for First-Class Mail parcels, Package Services parcels, and all Priority Mail mailpieces. Consult the USPS Domestic Mail Manual, Section 503 Extra Services (http://pe.usps.gov/text/dmm300/503.htm) for limitations on each method. Please note the following:

   - *Delivery Confirmation:* Delivery Confirmation will be included at no additional charge by default if MailClass is Priority, First, ParcelSelect, or CriticalMail. If you set DeliveryConfirmation to "OFF", your label will still contain a barcode, but USPS tracking will not be available. You may obtain limited tracking information using the StatusRequest Web method or through your account www.endicia.com.
     i. Delivery Confirmation for International Packages (E-USPS DELCON INTL) - Electronic Delivery Confirmation will now be available for international packages.
        - This is currently being marketed as a free service.

- Service will be available for the following mail categories: First-Class Package International Service (formerly First-Class Mail International) parcels, Priority Mail International Flat Rate Envelopes, and Small Flat Rate Boxes.

- Service will be available in January 2013 for items shipped to Canada. Additional countries will be added in April 2013.

- Labels will have markings to indicate that Electronic Delivery Confirmation Service is included.

- Delivery status on packages will be available on USPS.COM using "Track and Confirm" feature. Additionally, tracking will also be available on www.endicia.com/tracking and by using the "Requesting Package Status" API.

- *Signature Confirmation:* Can only be used with label type of Default. Not available for APO/FPO/DPO destinations.
- *Certified Mail:* The default value is ON for CertifiedMail label type; otherwise, it is OFF. Available for First-Class Letters, Flats and parcels and for Priority Mail. Certified Mail is the only tracking option available for some U.S. Territories and Possessions.

The USPS uses a Service Type Code (STC) to define all the combinations of services available for all mail classes. The STC is located in the 3rd through 5th positions in the tracking number (i.e., 94xxx...). The complete list can be found in USPS Publication 199. A sampling of STC's for First-Class Package Service and Priority Mail follows:

MailClass	DeliveryConfirmation	SignatureConfirmation	Certified	None
First	001	021	071	742
Priority	055	108	026	025

7. **Hold For Pickup Service.** The Hold For Pickup service allows mailpieces to be held at a designated Post Office location for pickup by a specified addressee or designee. The customer is notified by email or SMS text message (provided by the mailer or shipper as part of the label creation) that a package is available for pickup. This service provides the shipper with the date and time that the addressee took possession of the item. If the item has not been picked up within 5 days, the Post Office will make a second attempt to notify the addressee. The item will be returned to the sender if not picked up within 15 days. The Hold For Pickup service is available with Express Mail, Priority Mail (except Critical Mail), First-Class Mail parcels, and Parcel Select barcoded, non-presorted parcels.

   There is no charge for the Hold For Pickup service. Delivery Confirmation is included by default, or you may also select Signature Confirmation. USPS Insurance for a declared value greater than $200 can be optionally selected. No other services are available with Hold For Pickup service at this time.

   Label Server will return a Hold For Pickup facility. However, if you prefer to obtain that information yourself, you may use USPS Web Tools (www.usps.com/webtools) to obtain the following information which would then need to be passed along with the transaction information:

- HfpFacilityID
- HfpFacilityName
- HfpFacilityAddress1
- HfpFacilityCity
- HfpFacilityState
- HfpFacilityPostalCode
- HfpFacilityZIP4

Note: The Hold For Pickup service requires valid e-mail address or SMS notification number for the addressee. Otherwise, there is no way for the addressee to be notified that a package is awaiting pickup. Accordingly, Label Requests now include the following elements to support this:

- HfpEmailAddress
- HfpSMS

Label Server does not produce PS Form 3811, Domestic Return Receipt, at this time. PS Form 3811 must be manually completed and attached to the package with the label. These forms may be obtained from the US Postal Service. The following illustration is a Hold For Pickup label:

8. **Rubber Stamps.** A label can contain one or more user-supplied text labels that are called Rubber Stamps. The Endicia Label Server Web Service supports up to three Rubber Stamp fields within a label. Although each Rubber Stamp field can have as many as 50 characters, the actual number of characters that can be printed on the label depends on the content of the Rubber Stamp text. This is true for all labels which are not in EPL2 format because all text is drawn using proportional fonts. Labels in EPL2 format, however, use monospaced fonts. Therefore, the number of characters in the Rubber Stamp fields will be much less than the maximum allowed.

For example, a 4×6 default label in EPL2 format can accommodate 26 or fewer characters in each Rubber Stamp field. *Note that Express Mail and International labels will not print Rubber Stamp elements*.

9. **"To" and "From" Address Elements.** Due to minimum font size requirements and available space on various label formats, not all To and From address elements are available for every Label Type. Count each of the following as an individual line:

   ToName
   ToCompany
   ToAddress1
   ToAddress2
   ToAddress3
   ToAddress4
   ToCity, ToState, ToPostalCode, ToZIP4, ToDeliveryPoint *(Remember to account for space between elements)*
   ToCountryCode *(Required for International; returns the full country name on the label)*

   Key points:
   - Express Mail labels now support up to five "Ship To" address lines including the ToCity, ToState, and ToPostalCode line.
   - For all customs forms with 4x6c LabelSize, only three lines are available above the ToCity, ToState, ToPostalCode line, and the ToCountryCode line. So, you can use up to three of any combination of ToName, ToCompany, ToAddress1, and ToAddress2 to populate the complete address.
   - In addition, when printing 4x6c customs forms in EPL2 Image Format, the length of any of these lines must be 32 characters or fewer, and the combined ToCity, ToState, and ToPostal Code can be no more than 30 characters. The ZPLII Image Format allows up to 43 characters and 41 characters, respectively.

   Please thoroughly test the limitations with any new printers prior to beginning production to ensure that all lines and all characters print properly.

XML Request Code Samples

The following listing shows an example of the LabelRequest XML.

Listing 3-1 LabelRequest XML

```
<LabelRequest Test="YES" LabelType="Default" LabelSize="4X6"
ImageFormat="GIF">
  <RequesterID>abcd</RequesterID>
  <AccountID>123456</AccountID>
  <PassPhrase>samplePassPhrase</PassPhrase>
  <MailClass>Priority</MailClass>
  <DateAdvance>0</DateAdvance>
  <WeightOz>15</WeightOz>
  <MailpieceShape>Parcel</MailpieceShape>
  <Stealth>FALSE</Stealth>
  <Services InsuredMail="OFF" DeliveryConfirmation=”ON”
SignatureConfirmation="ON" />
  <Value>0</Value>
  <Description>Sample Label</Description>
  <PartnerCustomerID>12345ABCD</PartnerCustomerID>
```

```
  <PartnerTransactionID>6789EFGH</PartnerTransactionID>
  <ToName>Label Server Support</ToName>
  <ToCompany>DYMO Endicia</ToCompany>
  <ToAddress1>385 Sherman Ave.</ToAddress1>
  <ToCity>Palo Alto</ToCity>
  <ToState>CA</ToState>
  <ToPostalCode>94306</ToPostalCode>
  <ToZIP4>1864</ToZIP4>
  <ToDeliveryPoint>00</ToDeliveryPoint>
  <ToPhone>8005763279</ToPhone>
  <FromName>Benjamin Franklin</FromName>
  <ReturnAddress1>475 L'Enfant Plaza SW</ReturnAddress1>
  <FromCity>Washington</FromCity>
  <FromState>DC</FromState>
  <FromPostalCode>20260</FromPostalCode>
  <FromZIP4>0004</FromZIP4>
  <FromPhone>2025551212</FromPhone>
  <ResponseOptions PostagePrice="TRUE"/>
</LabelRequest>
```

LabelRequestResponse XML Syntax

The following table describes the XML tags for LabelRequestResponse.

Table 3-2: LabelRequestResponse XML Elements

Element	Type	Size or Values	Description
LabelRequestResponse	**Root**	**-**	**The root node for the XML response.**
Status	Numeric	0 Error code	Indicates if the label request succeeded or failed: Success Failed. The response contains an ErrorMessage element describing the error.
ErrorMessage	*Text*	*150*	*Error message. See Status (preceding entry).*
Base64LabelImage	Base64 string		Image of the label. If the Label node is present, then this element will not exist. See Note 1.
Label	*Node*		*Contains one or more <Image> elements for each part of the label.*
Image	Base64 string		First part of the label image.
PartNumber	Attribute, Numeric	1	
Image	*Base64 string*		*Second part of the label image.*
PartNumber	*Attribute, Numeric*	*2*	
:	*:*	*:*	*:*
Image	*Base64 string*		*nth (last) part of the label image.*
PartNumber	*Attribute,*	*N*	

Element	Type	Size or Values	Description
	Numeric		
CustomsForm	*Node*		*Contains one or more <Image> elements for each part of the customs form.* **Note:** *The CustomsForm node is retained for backward compatibility only.*
Image	Base64 string		First part of the customs form image.
PartNumber	Attribute, Numeric	1	
Image	*Base64 string*		*Second part of the customs form image.*
PartNumber	*Attribute, Numeric*	*2*	
:	:	:	:
Image	*Base64 string*		*nth (last) part of the customs form image.*
PartNumber	*Attribute, Numeric*	*N*	
TrackingNumber	Text	Variable Length	For domestic mail, this value is also copied to the PIC element. For international mail and domestic mail using customs forms, this value is also copied to the CustomsNumber element. **Note:** With the migration from Planet Code technology to IMb technology for DestinationConfirm services, the TrackingNumber returned can be up to 31 characters long.
PIC	*Text*		*Same as TrackingNumber.* *For domestic mail pieces only.*
CustomsNumber	*Text*	*13*	*Customs form number.*
FinalPostage	Currency	6.3	Total postage and fees for transaction.
TransactionID	Numeric		Unique ID for transaction - corresponds to the piece count for the account used.
TransactionDateTime	Date and Time	YYYYMMDDHHMMSS	Date and time the transaction was created. Expressed in UTC.
CostCenter	*Numeric*	*8*	*Cost-center code for accounting purposes.*
CostCenterAlphaNumeric	*Text*	*50*	*Cost-center code for accounting purposes.*

Element	Type	Size or Values	Description
			Note: *This element serves the same purpose as the CostCenter element above. Use this element if the cost center values are alphanumeric.Do not use both these elements together.*
ReferenceID	*Text*	*50*	*A reference value for logs.*
PostmarkDate	Date	YYYYMMDD	Indicates the postmark date.
PostageBalance	Currency	6.3	Remaining postage balance after label is generated, in dollars and cents (mils).
PostagePrice	*Node*		*Postage price for transaction.*
TotalAmount	*Attribute, Currency*	*6.3*	*Total amount of postage and fees. This value is also copied to the FinalPostage element.*
Postage	*Node*		
TotalAmount	*Attribute, Currency*	*6.3*	*Total amount of postage.*
MailService	*Text*	*50*	*Mail Service.*
Zone	*Text*	*0 (local zone), 1 through 8*	*Number of postal rate zones between the sender's and recipient's ZIP Codes.* *Applies only to domestic mail.*
IntraNDC	*Text*	*true* *false*	*Zone is Intra-NDC.* *Zone is Inter-NDC.* *Applies only to domestic mail.*
Pricing	*Text*	*Commercial* *CommercialBase* *CommercialPlus* *NSA* *Retail*	*Commercial pricing.* *Commercial Base pricing.* *Commercial Plus pricing.* *NSA pricing.* *Retail pricing.*
Fees	*Node*		
TotalAmount	*Attribute, Currency*	*6.3*	*Total amount of fees.*
CertificateOfMailing	*Currency*	*6.3*	*Certificate of Mailing fee. For future use.*
CertifiedMail	*Currency*	*6.3*	*Certified Mail fee.*
CollectOnDelivery	*Currency*	*6.3*	*Collect on Delivery (COD) fee. For future use.*
DeliveryConfirmation	*Currency*	*6.3*	*Delivery Confirmation fee.*
ElectronicReturnReceipt	*Currency*	*6.3*	*Electronic Return Receipt fee.*
InsuredMail	*Currency*	*6.3*	*Insured Mail fee.*
RegisteredMail	*Currency*	*6.3*	*Registered Mail fee. For future use.*

Element	Type	Size or Values	Description
RestrictedDelivery	*Currency*	*6.3*	*Restricted Delivery fee.*
ReturnReceipt	*Currency*	*6.3*	*Return Receipt fee.*
ReturnReceiptForMerchandise	*Currency*	*6.3*	*Return Receipt for Merchandise fee. For future use.*
SignatureConfirmation	*Currency*	*6.3*	*Signature Confirmation fee.*
SpecialHandling	*Currency*	*6.3*	*Special Handling fee. For future use.*
AdultSignature	*Currency*	*6.3*	*Adult Signature fee.*
AdultSignatureRestrictedDelivery	*Currency*	*6.3*	*Adult Signature Restricted Delivery fee.*
LiveAnimalSurcharge	*Currency*	*6.3*	*Live Animal Surcharge fee.*
The following elements are returned only if Services > HoldForPickup was selected in the Label Request:			
HfpFacilityID	*Text*	*50*	*Destination Facility ID.*
HfpFacilityName	*Text*	*50*	*Destination Facility Name.*
HfpFacilityAddress1	*Text*	*50*	*Destination Facility Address.*
HfpFacilityCity	*Text*	*50*	*Destination Facility City.*
HfpFacilityState	*Text*	*50*	*Destination Facility State.*
HfpFacilityPostalCode	*Text*	*25*	*Destination Facility 5-Digit ZIP Code.*
HfpFacilityZIP4	*Text*	*5*	*Destination Facility +4.*

Notes

1. **Base64 Image Locations.** Depending on your Label Request options, the Base64 images may be contained in different elements.

LabelType/MailClass	Base64LabelImage	Label/Image Node	CustomsForm/Image Node
Default: any domestic	✓		
Domestic		✓	
International		✓	
Default to military or territories/possessions w/<CustomsFormType>	✓		✓
International w/<CustomsFormType>			✓

Note: The <CustomsFormType> element and the CustomsForm/Image node are retained for backward-compatibility only. For more information, see Customs Form Options.

XML Response Code Samples

The following listing shows an example of the LabelRequestResponse XML.

Listing 3-2 LabelRequestResponse XML

```
<LabelRequestResponse>
  <Status>0</Status>
  <Base64LabelImage>
    .
    . (label image in base64 string format)
    .
  </Base64LabelImage>
```

```
  <PIC>94221234567891234567 83</PIC>
  <TrackingNumber>94221234567891234567 83</TrackingNumber>
  <FinalPostage>7.39</FinalPostage>
  <TransactionID>789</TransactionID>
  <TransactionDateTime>20101220060019</TransactionDateTime>
  <PostmarkDate>20080508</PostmarkDate>
  <PostageBalance>123.450</PostageBalance>
  <PostagePrice TotalAmount="7.39">
   <Postage TotalAmount="5.44">
    <MailService>Priority Mail</MailService>
    <Zone>4</Zone>
    <IntraNDC>FALSE</IntraNDC>
    <Pricing>CommercialBase</Pricing>
   </Postage>
   <Fees TotalAmount="1.95">
    <CertificateOfMailing>0</CertificateOfMailing>
    <CertifiedMail>0</CertifiedMail>
    <CollectOnDelivery>0</CollectOnDelivery>
    <DeliveryConfirmation>0</DeliveryConfirmation>
    <ElectronicReturnReceipt>0</ElectronicReturnReceipt>
    <InsuredMail>0</InsuredMail>
    <RegisteredMail>0</RegisteredMail>
    <RestrictedDelivery>0</RestrictedDelivery>
    <ReturnReceipt>0</ReturnReceipt>
    <ReturnReceiptForMerchandise>0</ReturnReceiptForMerchandise>
    <SignatureConfirmation>1.95</SignatureConfirmation>
    <SpecialHandling>0</SpecialHandling>
   </Fees>
  </PostagePrice>
</LabelRequestResponse>
```

4. BUYING POSTAGE API (REQUIRED)

RecreditRequest

To add postage to an account, use the BuyPostage or the BuyPostageXML Web method of the Endicia Label Server Web Service.

The following is a sample HTTP POST request for BuyPostageXML, which requires a RecreditRequest XML string as input. The **placeholders** shown need to be replaced with actual values:

```
POST /LabelService/EwsLabelService.asmx/BuyPostageXML HTTP/1.1
Host: www.envmgr.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

recreditRequestXML=<RecreditRequest> ... </RecreditRequest>
```

The response contains a RecreditRequestResponse XML string:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<RecreditRequestResponse> ... </RecreditRequestResponse>
```

RecreditRequest XML Syntax

The following table describes the XML tags for RecreditRequest.

Important notes about all elements:

- Omitting any value causes the Endicia Label Server Web Service to use the default value for that element.
- *Italic* items are optional.

Table 4-1: RecreditRequest XML Elements

Element	Type	Size or Values	Description
RecreditRequest	**Root**	**-**	**The root node for the XML request.**
RequesterID	Text	50	Requester ID (also called Partner ID) is used to uniquely identify the system making the request. This ID is assigned by Endicia.
RequestID	Text	50	Request ID to uniquely identify this Recredit request. This will be returned in response.
CertifiedIntermediary	Node		Certified Intermediary (CI) account authentication information.
AccountID	Numeric	6	Account ID for the Endicia postage account.

Element	Type	Size or Values	Description
PassPhrase	Text	64	Pass Phrase for the Endicia postage account.
RecreditAmount			Amount of postage, in dollars, to add to the account. Either use a predefined amount from the list or enter any value up to 99999.99. Defined amounts:
	Text	10	Add $10.00 to the account.
		25	Add $25.00 to the account.
		50	Add $50.00 to the account.
		100	Add $100.00 to the account.
		250	Add $250.00 to the account.
		500	Add $500.00 to the account.
		1000	Add $1,000.00 to the account.
		2500	Add $2,500.00 to the account.
		5000	Add $5,000.00 to the account.
		7500	Add $7,500.00 to the account.
		10000	Add $10,000.00 to the account.
		20000	Add $20,000.00 to the account.
	Currency	5.2	Any amount, at least $10.00 and up to $99,999.99, in unit of dollars and rounded to the nearest cent. The minimum amount of postage that can be purchased is $10. The maximum amount is based on the settings of the account.

The following listing shows an example of the RecreditRequest XML.

Listing 4-1: RecreditRequest XML

```
<RecreditRequest>
  <RequesterID>abcd</RequesterID>
  <RequestID>BP123</RequestID>
  <CertifiedIntermediary>
    <AccountID>123456</AccountID>
    <PassPhrase>samplePassPhrase</PassPhrase>
  </CertifiedIntermediary>
  <RecreditAmount>250</RecreditAmount>
</RecreditRequest>
```

RecreditRequestResponse XML Syntax

The following table describes the XML tags for RecreditRequestResponse.

Table 4-2: RecreditRequestResponse XML Elements

Element	Type	Size or Values	Description
RecreditRequestResponse	**Root**	**-**	**The root node for the XML response.**
Status	Numeric	0 Error code	Indicates if the request to buy postage succeeded or failed: Success Failed. The response contains an ErrorMessage element describing the error.
ErrorMessage	*Text*	*150*	*Error message. See Status (preceding entry).*
RequesterID	Text	50	Return of Requester ID—based on input value.
RequestID	Text	50	Return of Request ID—based on input value.
CertifiedIntermediary	Node		Certified Intermediary (CI) account status.
AccountID	Numeric	6	Account ID for the Endicia postage account.
SerialNumber	Numeric	12	The Serial Number of the last item created on the account.
PostageBalance	Currency	6.3	Remaining postage balance after re-credit in dollars and cents (mils)
AscendingBalance	Currency	6.3	Total amount of postage printed (after re-credit) in dollars and cents (mils)
AccountStatus	Text	A	Account Status flag: Active
DeviceID	Text	12	The account's 12 character Device ID (that appears in the indicium).

The following listing shows an example of a RecreditRequestResponse XML.

Listing 4-2: RecreditRequestResponse XML

```
<RecreditRequestResponse>
  <Status>0</Status>
  <RequesterID>abcd</RequesterID>
  <RequestID>BP123</RequestID>
  <CertifiedIntermediary>
    <AccountID>123456</AccountID>
    <SerialNumber>789</SerialNumber>
    <PostageBalance>76.55</PostageBalance>
    <AscendingBalance>123.45</AscendingBalance>
    <AccountStatus>A</AccountStatus>
    <DeviceID>071V00123456</DeviceID>
  </CertifiedIntermediary>
</RecreditRequestResponse>
```

Please note the following information about purchasing postage and bank-to-bank postage payment options.

Postage Purchases

- Purchase Methods: Postage purchases are billed when a customer purchases postage.
- Payment Types: Visa, MasterCard, American Express, Discover-Novus, Diner's Club International, Carte Blanche, checks, and direct debits (ACH) from US-based checking accounts, and ACH Credit issued to the USPS bank account will be accepted. Payment methods are subject to change at any time.
- When an account is initiated, the minimum postage purchase is $10, and the maximum is $500, and a customer's account may contain a maximum of $500 of postage at any given time. An unlimited number of purchases can be made. No sales taxes are applicable to purchases of postage.
  A customer may apply directly with Endicia's support staff (support@endicia.com) to have the postage purchase limits on an account increased after they have used the account and have established a postage buying pattern (usually a couple of weeks).

Endicia Bank-to-Bank Postage Payment Options

- ACH Withdrawal (Direct Debit):
  With the ACH Withdrawal method, you provide information about your checking account, and Endicia will withdraw the funds directly from your checking account and add the amount to your Endicia postage account balance. Endicia uses the USPS-authorized Federal Automated Clearing House (ACH) system to handle all payments. The purchase process in the client works exactly the same way with this method, and you can buy postage at any time to increase the balance in your Endicia account. If this payment method is used, your payment limits can be increased beyond the standard $500 restriction after an initial setup period.

  The following steps describe how to switch to direct debit:

  Steps

1. Log in to your online Endicia account at https://www.endicia.com/Account/LogIn/
2. Select **Update Profile > Payment Method**.
3. Select the **Deduct from my Checking Account…** option.
4. Enter your checking account and routing number.

Alternately, you can also contact support@endicia.com by email to switch.

- ACH Credit (or Push) to the US Postal Service:
  The ACH Credit (Push) method allows you to transfer funds directly from your bank to the USPS. The USPS then transfers those funds to Endicia to add to your account balance. The primary advantages to this method are: no bank account information needs to be provided to Endicia and any amount of postage may purchase. For customers that prefer to purchase in large amounts, this can be quite advantageous.

  The following steps describe how to use this method.

  Steps

  1. Instruct your bank to send an ACH TRANSFER (NOT A WIRE TRANSFER) to the USPS "lockbox" at Citibank.
     The account number is 30564608, and the Citibank ABA routing transit number is 021000089.
  2. Include your Endicia account number on the ACH Credit Transaction.
     Your bank may have a field for an "Invoice Number", "Order Number", "Individual or Company ID", or a note. If this information is omitted, your Endicia account cannot be credited for ten (10) business days or until another ACH transaction is made with your Endicia Account number in the proper field.
  3. Your bank may require additional verification of the transfer. The vendor name for the transfer is the "US Postal Service (Endicia)" and the tax ID is 41-0760000
  4. Expect the transfer to take 3-5 business to reach your Endicia account.

- Wire transfer to Endicia:
  You may institute a wire transfer of funds from your bank to the Endicia bank account at Union Bank of California, and Endicia will subsequently transfer these funds into the USPS bank account. Please include your Endicia account number in the wire tranfer notation field. Contact Endicia Accounting at accounting@endicia.com if you are interested in this method.
  Note: There is a $50 fee for each payment using this method.

- Pay by check:
  You may mail a check directly to Endicia to purchase postage. Make the check out to "Endicia Internet Postage", add your Endicia account number to the check or check stub, and mail to:

  PSI Systems, Inc.
  Attn: Accounting Department
  385 Sherman Ave.
  Palo Alto, CA 94306-1864

  The process to credit your account will take five business days after we receive the check.
  Note: There is a $50 fee for each payment using this method.

5. CHANGING THE PASS PHRASE API (REQUIRED)

ChangePassPhraseRequest

To change the Pass Phrase for an account, use the ChangePassPhrase or the ChangePassPhraseXML Web method of the Endicia Label Server Web Service.

The following is a sample HTTP POST request for ChangePassPhraseXML, which requires a ChangePassPhraseRequest XML string as input. The **placeholders** shown need to be replaced with actual values:

```
POST /LabelService/EwsLabelService.asmx/ChangePassPhraseXML HTTP/1.1
Host: www.envmgr.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

changePassPhraseRequestXML=<ChangePassPhraseRequest> ...
</ChangePassPhraseRequest>
```

The response contains a ChangePassPhraseRequestResponse XML string:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<ChangePassPhraseRequestResponse> ... </ChangePassPhraseRequestResponse>
```

ChangePassPhraseRequest XML Syntax

The following table describes the XML tags for ChangePassPhraseRequest.

Important notes about all elements:

- Omitting any value causes the Endicia Label Server Web Service to use the default value for that element.
- *Italic* items are optional.

Table 5-1: ChangePassPhraseRequest XML Elements

Element	Type	Size or Values	Description
ChangePassPhraseRequest	**Root**	**-**	**The root node for the XML request.**
RequesterID	Text	50	Requester ID (also known as Partner ID) is used to uniquely identify the system making the request. This ID is assigned by Endicia.
RequestID	Text	50	Request ID to uniquely identify this request. This will be returned in response.
CertifiedIntermediary	Node		Certified Intermediary (CI) account authentication information.

Element	Type	Size or Values	Description
AccountID	Numeric	6	Account ID for the Endicia postage account.
PassPhrase	Text	64	Pass Phrase for the Endicia postage account
NewPassPhrase	Text	64	New Pass Phrase for the Endicia postage account. Pass Phrase must be at least 5 characters long with a maximum of 64 characters. For added security, the Pass Phrase should be at least 10 characters long and include more than one word, use at least one uppercase and lowercase letter, one number and one non-text character (for example, punctuation). A Pass Phrase which has been used previously will be rejected.

The following listing shows an example of the ChangePassPhraseRequest XML.

Listing 5-1: ChangePassPhraseRequest XML

```
<ChangePassPhraseRequest>
  <RequesterID>abcd</RequesterID>
  <RequestID>CPP123</RequestID>
  <CertifiedIntermediary>
   <AccountID>123456</AccountID>
   <PassPhrase>samplePassPhrase</PassPhrase>
  </CertifiedIntermediary>
  <NewPassPhrase>newPassPhrase</NewPassPhrase>
</ChangePassPhraseRequest>
```

ChangePassPhraseRequestResponse XML Syntax

The following table describes the XML tags for ChangePassPhraseRequestResponse.

Table 5-2: ChangePassPhraseRequestResponse XML Elements

Element	Type	Size or Values	Description
ChangePassPhraseRequestResponse	**Root**	**-**	**The root node for the XML response.**
Status	Numeric	0 Error code	Indicates if the request to change Pass Phrase succeeded or failed: Success Failed. The response contains an ErrorMessage element describing the error.
ErrorMessage	*Text*	*150*	*Error message. See Status (preceding entry).*
RequesterID	Text	50	Return of Requester ID—based on input value.
RequestID	Text	50	Return of Request ID—based on input value.

The following listing shows an example of the ChangePassPhraseRequestResponse XML.

Listing 5-2: ChangePassPhraseRequestResponse XML

```
<ChangePassPhraseRequestResponse>
  <Status>0</Status>
  <RequesterID>abcd</RequesterID>
  <RequestID>CPP123</RequestID>
</ChangePassPhraseRequestResponse>
```

6. RESETTING SUSPSENDED ACCOUNT STATUS API (OPTIONAL)

The following two Web methods can be used to reset the status of an account that has been suspended due to multiple failed attempts to enter the correct Pass Phrase.

Note: This API should *only* be used for accounts that have been *suspended* due to multiple bad login attempts. For changing the Pass Phrase of an active account, use the Change PassPhrase API.

Part 1: GetChallengeQuestion

Use the GetChallengeQuestionXML Web method to initiate the request for resetting a suspended account.

The following is a sample HTTP POST request for GetChallengeQuestionXML, which requires a ChallengeQuestion XML string as input. The **placeholders** shown need to be replaced with actual values:

```
POST /LabelService/EwsLabelService.asmx/GetChallengeQuestionXML HTTP/1.1
Host: www.envmgr.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

ChallengeQuestionRequestXML=<ChallengeQuestionRequest> ...
</ChallengeQuestionRequest>
```

The response contains a ChallengeQuestionResponse XML string:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
ChallengeQuestionResponseXML=<ChallengeQuestionResponse> ...
</ChallengeQuestionResponse>
```

GetChallengeQuestion XML Syntax

The following table describes the XML tags for GetChallengeQuestionRequest.

Important notes about all elements:

- Omitting any value causes the Endicia Label Server Web Service to use the default value for that element.
- *Italic* items are optional.

Table 6-1: ChallengeQuestionRequest XML Elements

Element	Type	Size or Values	Description
ChallengeQuestionRequest	**Root**	**-**	**The root node for the XML request.**
RequesterID	Text	50	Requester ID (also known as Partner ID) is used to uniquely identify the system making the request. This ID is assigned by Endicia.

Element	Type	Size or Values	Description
RequestID	Text	50	Request ID to uniquely identify this request. This will be returned in response.
AccountID	Numeric	6	Account ID for the Endicia postage account.
EMail	Text	64	Email ID on File

The following listing shows an example of the GetChallengeQuestionRequest XML.

Listing 6-1: GetChallengeRequest XML

```
<ChallengeQuestionRequest>
  <RequesterID>abcd</RequesterID>
  <RequestID>CPP123</RequestID>
  <AccountID>123456</AccountID>
  <EMail>JohnDoe@Company.com</EMail>
</ChallengeQuestionRequest>
```

ChallengeQuestionResponse XML Syntax

The following table describes the XML tags for ChallengeQuestionResponse.

Table 6-2: ChallengeQuestionResponse XML Elements

Element	Type	Size or Values	Description
ChallengeQuestionResponse	**Root**	**-**	**The root node for the XML response.**
Question	Text	50	Challenge Question on file for the Endicia postage account.
Status	Numeric	 0 Error code	Indicates if the request to get the challenge question on file succeeded or failed: Success Failed. The response contains an ErrorMessage element describing the error.
ErrorMessage	*Text*	*150*	*Error message. See Status (preceding entry).*
RequesterID	Text	50	Return of Requester ID—based on input value.
RequestID	Text	50	Return of Request ID—based on input value.

The following listing shows an example of the ChallengeQuestionResponse XML.

Listing 6-2: GetChallengeQuestionResponse XML

```
<ChallengeQuestionResponse>
  <Question>What is your favorite color</Question>
  <Status>0</Status>
  <RequesterID>abcd</RequesterID>
  <RequestID>CPP123</RequestID>
</ChallengeQuestionResponse>
```

Part 2: ResetSuspendedAccount

Use the ResetSuspendedAccount XML Web method to complete the request for resetting the status of a suspended account.

The following is a sample HTTP POST request for ResetSuspendedAccountRequestXML, which requires a ResetSuspendedAccountRequest XML string as input. The **placeholders** shown need to be replaced with actual values:

```
POST /LabelService/EwsLabelService.asmx/ResetSuspendedAccountXML HTTP/1.1
Host: www.envmgr.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

ResetSuspendedAccountRequestXML=<ResetSuspendedAccountRequest> ...
</ResetSuspendedAccountRequest>
```

The response contains a ResetSuspendedAccountResponse XML string:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
ResetSuspendedAccountRequestResponseXML=<ResetSuspendedAccountRequestRespo
nse> ...
</ResetSuspendedAccountRequestResponse>
```

ResetSuspendedAccountRequest XML Syntax

The following table describes the XML tags for ResetSuspendedAccount Request.

Important notes about all elements:

- Omitting any value causes the Endicia Label Server Web Service to use the default value for that element.
- *Italic* items are optional.

Table 6-3: ResetSuspendedAccountRequest XML Elements

Element	Type	Size or Values	Description
ResetSuspendedAccountRequest	**Root**	**-**	**The root node for the XML request.**
RequesterID	Text	50	Requester ID (also known as Partner ID) is used to uniquely identify the system making the request. This ID is assigned by Endicia.
RequestID	Text	50	Request ID to uniquely identify this request. This will be returned in response.
AccountID	Numeric	6	Account ID for the Endicia postage account.
ChallengeAnswer	Text	50	Answer to the challenge question on file for the Endicia postage account.
NewPassPhrase	Text	64	New Pass Phrase for the Endicia postage account. The Pass Phrase must be at least 5 characters long with a maximum of 64 characters. For added security, the Pass Phrase should be at least 10 characters long and include more than one word, use at least one uppercase and lowercase letter, one number, and one non-text character (for example, punctuation). A Pass Phrase which has been used previously will be rejected.

The following listing shows an example of the ResetSuspendedAccountRequest XML.

Listing 6-3: ResetSuspendedAccountRequest XML

```
<ResetSuspendedAccountRequest>
  <RequesterID>abcd</RequesterID>
  <RequestID>CPP123</RequestID>
  <AccountID>123456</AccountID>
  <ChallengeAnswer>Black</ChallengeAnswer>
  <NewPassPhrase>Endicia</NewPassPhrase>
</ResetSuspendedAccountRequest>
```

ResetSuspendedAccountRequestResponse XML Syntax

The following table describes the XML tags for ResetSuspendedAccountRequestResponse.

Table 6-4: ResetSuspendedAccountRequestResponse XML Elements

Element	Type	Size or Values	Description
ResetSuspendedAccountRequestResponse	**Root**	**-**	**The root node for the XML response.**
Status	Numeric	 0 Error code	Indicates if the request to reset the status of the suspended account succeeded or failed: Success Failed. The response contains an ErrorMessage element describing the error.
ErrorMessage	*Text*	*150*	*Error message. See Status (preceding entry).*
RequesterID	Text	50	Return of Requester ID—based on input value.
RequestID	Text	50	Return of Request ID—based on input value.

The following listing shows an example of the ResetSuspendedAccountRequestResponse XML.

Listing 6-4: ResetSuspendedAccountRequestResponse XML

```
<ResetSuspendedAccountRequestResponse>
  <Status>0</Status>
  <RequesterID>abcd</RequesterID>
  <RequestID>CPP123</RequestID>
</ResetSuspendedAccountRequestResponse>
```

7. INTRODUCTION TO CALCULATING POSTAGE AND FEES

The following API options for calculating postage and fees in Label Server are:

- PostageRateRequest returns the rate for a single MailClass.
- PostageRatesRequest returns the rate for multiple classes of mail.

You can use the single request to obtain postage and fees information when you know which class of mail you will be using and will need to obtain the rate and fees first.

You can use the multiple rates request to obtain the best rate for a particular package.
The following table summarizes how you determine which option to use:

Table 7-1: Options for Calculating Postage and Fees

Option Type	PostageRateRequest	PostageRatesRequest
Price for a known class of service	Y	N
Return class options with price	N	Y
Can return a retail price	Y*	N**

* Choose Pricing to return requested rate.
** Returns pricing level associated with the customer account number, except for Library Mail, Media Mail, and Standard Post, which are always retail.

When using either of these API's, remember that you must explicitly declare each service you are requesting or that service will not be included in the TotalAmount element or the fees node of the response. For example, if you do not include a tracking method (DeliveryConfirmation, CertifiedMail, or SignatureConfirmation) in a Label Request, DeliveryConfirmation will be selected as the default tracking method, and the associated fee will be included in the Postage and Fees for that label. However, if you do not include a tracking method in a rate or rates request, the Label Server will not include it in the response.

Note also that services for which the USPSdoes not charge a fee are not available options in these API's.

8. CALCULATE POSTAGE AND FEES API (SINGLE)

PostageRateRequest

To calculate the postage and fees for a single mailpiece, use the CalculatePostageRate or the CalculatePostageRateXML Web method of the Endicia Label Server Web Service.

The following is a sample HTTP POST request for CalculatePostageRateXML, which requires a PostageRateRequest XML string as input. You must replace the **placeholders** with actual values.

```
POST /LabelService/EwsLabelService.asmx/CalculatePostageRateXML HTTP/1.1
Host: www.envmgr.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

postageRateRequestXML=<PostageRateRequest> ... </PostageRateRequest>
```

The response contains a PostageRateResponse XML string:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<PostageRateResponse> ... </PostageRateResponse>
```

PostageRateRequest XML Syntax

The following table describes the XML tags for PostageRateRequest.

Important notes about all elements:

- Omitting any value causes the Endicia Label Server Web Service to use the default value for that element.
- *Italic* items are optional.

Table 8-1: PostageRateRequest XML Elements

Element	Type	Size or Values	Description
PostageRateRequest	**Root**	**-**	**The root node for the XML request.**
RequesterID	Text	50	Requester ID (also called Partner ID) is used to uniquely identify the system making the request. This ID is assigned by Endicia.
CertifiedIntermediary	Node		Certified Intermediary (CI) account authentication information.
AccountID	Numeric	6	Account ID for the Endicia postage account.
PassPhrase	Text	64	Pass Phrase for the Endicia

Element	Type	Size or Values	Description
			postage account.
MailClass	Text	Express First LibraryMail MediaMail StandardPost ParcelSelect Priority CriticalMail ExpressMailInternational FirstClassMailInternational FirstClassPackageInternationalService PriorityMailInternational	Domestic: Express Mail First-Class Package Service and First-Class Mail Parcel Library Mail Media Mail Standard Post (Formerly Parcel Post) Parcel Select/Parcel Select Lightweight Priority Mail Critical Mail International: Express Mail International First-Class Mail International First-Class Package International Service Priority Mail International
DateAdvance	*Numeric*	*0-7 (default range)*	*The number of days to advance date on the indicium.* *When used, this element advances the date on the indicium by the number of days selected from the range of values.* *Default range: (0-7) days.*
Pricing	*Text*	*CommercialBase* *CommercialPlus* *Retail*	*Commercial Base pricing.* *Commercial Plus pricing.* *Retail pricing.* *If this element is not supplied, pricing will be based on the MailClass and any qualified discounts available to AccountID.* **Important!** If this element is set to CommercialPlus, but the account does not qualify for such pricing, the Web method will return an error.
WeightOz	Numeric	4.1	Weight of the package, in ounces.
MailpieceShape	*Text*	*Card* *Letter* *Flat* *Parcel* *LargeParcel* *IrregularParcel*	*Shape of the mailpiece.* *See the LabelType, LabelSize and MailpieceShape by MailClass for more information.*

Element	Type	Size or Values	Description
		FlatRateEnvelope *FlatRateLegalEnvelope* *FlatRatePaddedEnvelope* *FlatRateGiftCardEnvelope* *FlatRateWindowEnvelope* *FlatRateCardboardEnvelope* *SmallFlatRateEnvelope* *SmallFlatRateBox* *MediumFlatRateBox* *LargeFlatRateBox* *DVDFlatRateBox* *LargeVideoFlatRateBox* *RegionalRateBoxA* *RegionalRateBoxB* *RegionalRateBoxC* *LargeFlatRateBoardGameBox* Containers for Open & Distribute services: *HalfTrayBox* *FullTrayBox* *EMMTrayBox* *FlatTubTrayBox*	*PMOD container-specific pricing available.*
MailpieceDimensions	*Node*		*Dimensions of the mailpiece.* *Required for calculating Cubic pricing and for Priority Mail mailpieces going to Zones 5-8 which are over 1 cubic foot.* *Ignored if MailpieceShape is set to LargeParcel.* *All values must be in inches.*
Length	Numeric	3.3	Length of the mailpiece. For best rates, always enter the longest dimension as the length.
Width	Numeric	3.3	Width of the mailpiece.
Height	Numeric	3.3	Height (or thickness) of the mailpiece.
AutomationRate	*Text*	*TRUE* *FALSE*	*Use applicable automation rate for selected mail class.* *Use retail price. (Default)* *Available only for letter shape mailpiece using First-Class.*
PackageTypeIndicator	*Text*	*Null*	*Package Type is regular (Default)*

Element	Type	Size or Values	Description
		Softpack	*Commercial Plus cubic price for soft pack packaging alternative*
Machinable	*Text*	*TRUE* *FALSE*	*Package is machinable. (Default)* *Package is non-machinable.* *If a Parcel Select mailpiece marked as machinable is over 35 lbs. in weight, it will automatically be charged the non-machinable rate.*
ServiceLevel	*Text*	*NextDay2ndDayPOToAddressee*	*Next Day or Second Day Post Office to Addressee Service* *Applies only to Express Mail. For more information, see Note 3 in* Requesting a Shipping Label API (Required)
SundayHolidayDelivery	*Text*	 *TRUE* *FALSE* *SUNDAY* *HOLIDAY*	*For Express Mail only:* *Request Sunday and Holiday Delivery Service.* *Do not deliver on Sunday or holiday. (Default)* *Request Sunday Delivery Service* *Request Holiday Delivery Service*
SortType	*Text*	*NDC* *FiveDigit* *MixedNDC* *Nonpresorted* *Presorted* *SCF* *SinglePiece* *ThreeDigit*	*Sort level for applicable mail classes.* *Required for Parcel Select and Parcel Select LightWeight.* *Defaults to SinglePiece for mailpieces which do not require a sort type.*
LiveAnimalSurcharge	*Text*	 *TRUE* *FALSE*	*Used only for shipping live animals, such as day-old chicks.* *Surcharge is applied. Surcharge is based on weight and is only added if the destination zone is greater than 4.* *Surcharge is not applied (Default).*
Services	*Node*		*Special Services requested for the package.*
CertifiedMail	*Attribute*	*OFF*	*Certified Mail not requested.*

Element	Type	Size or Values	Description
		ON	*Certified Mail requested.*
COD	*Attribute*	*OFF* *ON*	*COD not requested.* *COD requested.*
DeliveryConfirmation	*Attribute*	*OFF* *ON*	*Delivery Confirmation not requested. (Default)* *Delivery Confirmation requested.*
ElectronicReturnReceipt	*Attribute*	*OFF* *ON*	*Electronic Return Receipt not requested. (Default)* *Electronic Return Receipt requested.*
InsuredMail	*Attribute*	*OFF* *UspsOnline* *ENDICIA*	*No insurance requested. (Default)* *USPS Online Insurance requested.* *Endicia Insurance requested (Maximum insurable value: $10,000)* *USPS Online Insurance is available only for mailpieces with Delivery or Signature Confirmation.* *USPS Insurance is not allowed for International Mail.* *Endicia insurance fee is not included in the postage price. It is billed to your account.*
RestrictedDelivery	*Attribute*	*OFF* *ON*	*Restricted Delivery not requested. (Default)* *Restricted Delivery requested.*
ReturnReceipt	*Attribute*	*OFF* *ON*	*Return Receipt not requested. (Default)* *Return Receipt requested.*
SignatureConfirmation	*Attribute*	*OFF* *ON*	*Signature Confirmation not requested. (Default)* *Signature Confirmation requested.*
AdultSignature	*Attribute*	*OFF* *ON*	*Adult Signature not requested. (Default)* *Adult Signature requested.*

Element	Type	Size or Values	Description
AdultSignatureRestrictedDelivery	*Attribute*	*OFF* *ON*	*Adult Signature Restricted Delivery not requested. (Default)* *Adult Signature Resticted Delivery requested.*
Value	*Currency*	*5.2*	*Value of the item(s) in the mailpiece.*
CODAmount	*Currency*	*5.2*	*COD amount to collect.* *Required if COD is requested for the mailpiece.*
InsuredValue	*Currency*	*5.2*	*Insured value of the mailpiece.* *Required if insurance or COD is requested for the mailpiece.*
EntryFacility	*Text*	*DNDC* *DDU* *DSCF* *ONDC* *Other*	*Postal facility where the mail is entered. The Zip Code of this facility must be specified in POZipCode.* *Required for Parcel Select and Parcel Select LightWeight.* *Destination NDC* *Destination Delivery Unit* *Destination Sectional Center Facility* *Origin NDC* *Other postal facility (Default)*
FromPostalCode	Text	5	Sender's postal code. The format is ZIP5. *For Parcel Select and Parcel Select LightWeight, the value of this element contains the Zip Code of the postal facility specified in EntryFacility.*
ToPostalCode	Text	5 15	Recipient's postal code. For Domestic Mail, the format is ZIP5 (required). For International Mail (optional).
ToCountryCode	*Text*	*2*	*Two character country code of the recipient's country.* *Refer to the Country Code Table for a list of valid country codes.* Required for International Mail.
ShipDate	*Date*	*MM/DD/YYYY*	*Date mailpiece is shipped.* *Required for Express Mail*

Element	Type	Size or Values	Description
			Sunday and Holiday Delivery Service. Ignored for other mail classes.
ShipTime	*Time*	*HH:MM AM or HH:MM PM*	*Time mailpiece is shipped.* *Applies only to Express Mail Sunday and Holiday Delivery Service. Ignored for other mail classes.* *If this element is not supplied, it defaults to 12:01 AM.*
IsConsolidator	*Text*	*TRUE* *FALSE*	*Rate Consolidator "no-postage" Label* *Do not rate Consolidator "no-postage" label (Default)*
ResponseOptions	*Node*		*Optional XML elements to include in the PostageRateResponse.*
PostagePrice	*Attribute*	*TRUE* *FALSE*	*TRUE means the response contains the PostagePrice node.*

The following listing shows an example of the PostageRateRequest XML.

Listing 8-1: PostageRateRequest XML

```
<PostageRateRequest>
  <RequesterID>abcd</RequesterID>
  <CertifiedIntermediary>
   <AccountID>123456</AccountID>
   <PassPhrase>samplePassPhrase</PassPhrase>
  </CertifiedIntermediary>
  <MailClass>Priority</MailClass>
  <WeightOz>8.7</WeightOz>
  <MailpieceShape>Parcel</MailpieceShape>
  <Machinable>True</Machinable>
  <Services DeliveryConfirmation="OFF" SignatureConfirmation="ON"/>
  <FromPostalCode>95747</FromPostalCode>
  <ToPostalCode>94301</ToPostalCode>
  <ResponseOptions PostagePrice="TRUE"/>
</PostageRateRequest>
```

PostageRateResponse XML Syntax

The following table describes the XML tags for PostageRateResponse.

Table 8-2: PostageRateResponse XML Elements

Element	Type	Size or Values	Description
PostageRateResponse	**Root**	**-**	**The root node for the XML response.**
Status	Numeric	0 Error code	Indicates if the request to calculate postage succeeded or failed: Success Failed. The response contains an ErrorMessage element describing the error.
ErrorMessage	*Text*	*150*	*Error message. See Status (preceding entry).*
RequesterID	Text	50	Return of Requester ID - based on input value.
Zone	*Text*	*0 (local zone), 1 through 8*	*Number of postal rate zones between the sender's and recipient's ZIP Codes.* *Applies only to domestic mail.*
Postage	Node		Postage price for requested mail service. The response will not contain this node if PostagePrice in ResponseOptions is set to TRUE.
MailService	Text	50	Mail Service.
Rate	Currency	6.3	Calculated postage price including all fees.
PostagePrice	*Node*		*Postage price for requested Mail Service.* *The response contains this node only if PostagePrice in ResponseOptions is set to TRUE.*
TotalAmount	*Attribute, Currency*	*6.3*	*Total amount of postage and fees.*
Postage	*Node*		
TotalAmount	*Attribute, Currency*	*6.3*	*Total amount of postage.*
MailService	*Text*	*50*	*Mail Service.*
Zone	*Text*	*0 (local zone), 1 through 8*	*Number of postal rate zones between the sender's and recipient's ZIP Codes.* *Applies only to domestic mail.*
IntraNDC	*Text*	*true*	*Zone is Intra-NDC.*

Element	Type	Size or Values	Description
		false	*Zone is Inter-NDC.* *Applies only to domestic mail.*
Pricing	*Text*	*Commercial* *CommercialBase* *CommercialPlus* *NSA* *Retail*	*Commercial pricing.* *Commercial Base pricing.* *Commercial Plus pricing.* *NSA pricing.* *Retail pricing.*
Fees	*Node*		
TotalAmount	*Attribute, Currency*	*6.3*	*Total amount of fees.*
CertificateOfMailing	*Currency*	*6.3*	*Certificate of Mailing fee.* *For future use.*
CertifiedMail	*Currency*	*6.3*	*Certified Mail fee.*
CollectOnDelivery	*Currency*	*6.3*	*Collect on Delivery (COD) fee.* *For future use.*
DeliveryConfirmation	*Currency*	*6.3*	*Delivery Confirmation fee.*
ElectronicReturnReceipt	*Currency*	*6.3*	*Electronic Return Receipt fee.*
InsuredMail	*Currency*	*6.3*	*Insured Mail fee.*
RegisteredMail	*Currency*	*6.3*	*Registered Mail fee.*
RestrictedDelivery	*Currency*	*6.3*	*Restricted Delivery fee.* *For future use.*
ReturnReceipt	*Currency*	*6.3*	*Return Receipt fee.*
ReturnReceiptForMerchandise	*Currency*	*6.3*	*Return Receipt for Merchandise fee.* *For future use.*
SignatureConfirmation	*Currency*	*6.3*	*Signature Confirmation fee.*
SpecialHandling	*Currency*	*6.3*	*Special Handling fee.* *For future use.*
AdultSignature	*Currency*	*6.3*	*Adult Signature fee.*
AdultSignatureRestrictedDelivery	*Currency*	*6.3*	*Adult Signature Restricted Delivery fee.*
LiveAnimalSurcharge	*Currency*	*6.3*	*Live Animal Surcharge fee.*

The following listing shows an example of the PostageRateResponse XML.

Listing 8-2: PostageRateResponse XML

```
<PostageRateResponse>
  <Status>0</Status>
  <Zone>2</Zone>
  <PostagePrice TotalAmount="7.39">
    <Postage TotalAmount="5.44">
      <MailService>Priority Mail</MailService>
      <Zone>2</Zone>
      <IntraNDC>TRUE</IntraNDC>
      <Pricing>CommercialBase</Pricing>
    </Postage>
    <Fees TotalAmount="1.95">
      <CertificateOfMailing>0</CertificateOfMailing>
      <CertifiedMail>0</CertifiedMail>
      <CollectOnDelivery>0</CollectOnDelivery>
      <DeliveryConfirmation>0</DeliveryConfirmation>
      <ElectronicReturnReceipt>0</ElectronicReturnReceipt>
      <InsuredMail>0</InsuredMail>
      <RegisteredMail>0</RegisteredMail>
      <RestrictedDelivery>0</RestrictedDelivery>
      <ReturnReceipt>0</ReturnReceipt>
      <ReturnReceiptForMerchandise>0</ReturnReceiptForMerchandise>
      <SignatureConfirmation>1.95</SignatureConfirmation>
      <SpecialHandling>0</SpecialHandling>
    </Fees>
  </PostagePrice>
</PostageRateResponse>
```

9. CALCULATE POSTAGE AND FEES API (MULTIPLE)

PostageRatesRequest

To calculate the postage and fees for multiple mail classes for a mailpiece, use the CalculatePostageRates or the CalculatePostageRatesXML Web method of the Endicia Label Server Web Service.

The following is a sample HTTP POST request for CalculatePostageRatesXML, which requires a PostageRatesRequest XML string as input. You must replace the **placeholders** with actual values.

```
POST /LabelService/EwsLabelService.asmx/CalculatePostageRatesXML HTTP/1.1
Host: www.envmgr.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

postageRatesRequestXML=<PostageRatesRequest> ... </PostageRatesRequest>
```

The response contains a PostageRatesResponse XML string:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<PostageRatesResponse> ... </PostageRatesResponse>
```

PostageRatesRequest XML Syntax

The following table describes the XML tags for PostageRatesRequest.

Important notes about all elements:

- Omitting any value causes the Endicia Label Server Web Service to use the default value for that element.
- *Italic* items are optional.

Table 9-1: PostageRatesRequest XML Elements

Element	Type	Size or Values	Description
PostageRatesRequest	**Root**	**-**	**The root node for the XML request.**
RequesterID	Text	50	Requester ID (also called Partner ID) is used to uniquely identify the system making the request. This ID is assigned by Endicia.
CertifiedIntermediary	Node		Certified Intermediary (CI) account authentication information.
AccountID	Numeric	6	Account ID for the Endicia postage account.

Element	Type	Size or Values	Description
PassPhrase	Text	64	Pass Phrase for the Endicia postage account.
MailClass	Text	Domestic International	Calculate rates for all applicable domestic mail classes. Calculate rates for all applicable international mail classes.
DateAdvance	*Numeric*	*0-7 (default range)*	*The number of days to advance date on the indicium.* *When used, this element advances the date on the indicium by the number of days selected from the range of values.* *Default range: (0-7) days.*
WeightOz	Numeric	4.1	Weight of the package, in ounces.
MailpieceShape	*Text*	*Card* *Letter* *Flat* *Parcel* *LargeParcel* *IrregularParcel* *FlatRateEnvelope* *FlatRateLegalEnvelope* *FlatRatePaddedEnvelope* *FlatRateGiftCardEnvelope* *FlatRateWindowEnvelope* *FlatRateCardboardEnvelope* *SmallFlatRateEnvelope* *SmallFlatRateBox* *MediumFlatRateBox* *LargeFlatRateBox* *DVDFlatRateBox* *LargeVideoFlatRateBox* *RegionalRateBoxA* *RegionalRateBoxB* *LargeFlatRateBoardGameBox* Containers for Open & Distribute services: *HalfTrayBox* *FullTrayBox*	*Shape of the mailpiece.* *See the LabelType, LabelSize and MailpieceShape by MailClass for more information.* *PMOD Container-specific pricing available.*

Element	Type	Size or Values	Description
		EMMTrayBox *FlatTubTrayBox*	
MailpieceDimensions	*Node*		*Dimensions of the mailpiece.* *Required for calculating Cubic pricing and for Priority Mail mailpieces going to Zones 5-8 which are over 1 cubic foot.* *Ignored if MailpieceShape is set to LargeParcel.* *All values must be in inches.*
Length	Numeric	3.3	Length of the mailpiece. For best rates, always enter the longest dimension as the length.
Width	Numeric	3.3	Width of the mailpiece.
Height	Numeric	3.3	Height (or thickness) of the mailpiece.
Machinable	*Text*	*TRUE* *FALSE*	*Package is machinable. (Default)* *Package is non-machinable.*
LiveAnimalSurcharge	*Text*	 *TRUE* *FALSE*	*Used only for shipping live animals, such as day-old chicks.* *Surcharge is applied. Surcharge is based on weight and is only added if the destination zone is greater than 4.* *Surcharge is not applied (Default).*
Services	*Node*		*Special Services requested for the package.*
CertifiedMail	*Attribute*	*OFF* *ON*	*Certified Mail not requested. (Default)* *Certified Mail requested.*
COD	*Attribute*	*OFF* *ON*	*COD not requested. (Default)* *COD requested.*

Element	Type	Size or Values	Description
DeliveryConfirmation	*Attribute*	*OFF* *ON*	*Delivery Confirmation not requested. (Default)* *Delivery Confirmation requested.*
ElectronicReturnReceipt	*Attribute*	*OFF* *ON*	*Electronic Return Receipt not requested. (Default)* *Electronic Return Receipt requested.*
InsuredMail	*Attribute*	*OFF* *UspsOnline* *ENDICIA*	*No insurance requested. (Default)* *USPS Online Insurance requested.* *Endicia Insurance requested (Maximum insurable value: $10,000)* *USPS Online Insurance is available only for mailpieces with Delivery or Signature Confirmation.* *USPS Insurance is not allowed for International Mail.* *Endicia insurance fee is not included in the postage price. It is billed to your account.*
RestrictedDelivery	*Attribute*	*OFF* *ON*	*Restricted Delivery not requested. (Default)* *Restricted Delivery requested.*
ReturnReceipt	*Attribute*	*OFF* *ON*	*Return Receipt not requested. (Default)* *Return Receipt requested.*
SignatureConfirmation	*Attribute*	*OFF* *ON*	*Signature Confirmation not requested. (Default)* *Signature Confirmation requested.*
AdultSignature	*Attribute*	*OFF* *ON*	*Adult Signature not requested. (Default)* *Adult Signature requested.*

Element	Type	Size or Values	Description
AdultSignatureRestrictedDelivery	*Attribute*	*OFF* *ON*	*Adult Signature Restricted Delivery not requested. (Default)* *Adult Signature Resticted Delivery requested.*
CODAmount	*Currency*	*5.2*	*COD amount to collect.* *Required if COD is requested for the mailpiece.*
InsuredValue	*Currency*	*5.2*	*Insured value of the mailpiece.* *Required if insurance or COD is requested for the mailpiece.*
FromPostalCode	Text	5	Sender's postal code. The format is ZIP5.
ToPostalCode	Text	 5 15	Recipient's postal code. For Domestic Mail, the format is ZIP5 (required). For International Mail (optional).
ToCountryCode	*Text*	*2*	*Two character country code of the recipient's country.* *Refer to the Country Code Table for a list of valid country codes.* Required for International Mail.

The following listing shows an example of the PostageRatesRequest XML.

Listing 9-1: PostageRatesRequest XML

```
<PostageRatesRequest>
  <RequesterID>abcd</RequesterID>
  <CertifiedIntermediary>
   <AccountID>123456</AccountID>
   <PassPhrase>samplePassPhrase</PassPhrase>
  </CertifiedIntermediary>
  <MailClass>Domestic</MailClass>
  <WeightOz>8.7</WeightOz>
  <MailpieceShape>Parcel</MailpieceShape>
  <Machinable>True</Machinable>
  <Services DeliveryConfirmation="OFF" SignatureConfirmation="ON"/>
  <FromPostalCode>94306</FromPostalCode>
  <ToPostalCode>20260</ToPostalCode>
</PostageRatesRequest>
```

PostageRatesResponse XML Syntax

The following table describes the XML tags for PostageRatesResponse.

Table 9-2: PostageRatesResponse XML Elements

Element	Type	Size or Values	Description
PostageRatesResponse	**Root**	**-**	**The root node for the XML response.**
Status	Numeric	0 Error code	Indicates if the request to calculate postage succeeded or failed: Success Failed. The response contains an ErrorMessage element describing the error.
ErrorMessage	*Text*	*150*	*Error message. See Status (preceding entry).*
RequesterID	Text	50	Return of Requester ID - based on input value.
PostagePrice	*Node*		*Postage price for requested Mail Service.* There may be multiple instances of this node for every applicable Mail Service.
TotalAmount	*Attribute, Currency*	*6.3*	*Total amount of postage and fees.*
Postage	*Node*		
TotalAmount	*Attribute, Currency*	*6.3*	*Total amount of postage.*
MailService	*Text*	*50*	*Mail Service.*
Zone	*Text*	*0 (local zone), 1 through 8*	*Number of postal rate zones between the sender's and recipient's ZIP Codes.* *Applies only to domestic mail.*
IntraNDC	*Text*	*True* *false*	*Zone is Intra-NDC.* *Zone is Inter-NDC.* *Applies only to domestic mail.*
Pricing	*Text*	*Commercial* *CommercialBase* *CommercialPlus* *NSA* *Retail*	*Commercial pricing.* *Commercial Base pricing.* *Commercial Plus pricing.* *NSA pricing.* *Retail pricing.*
Fees	*Node*		
TotalAmount	*Attribute, Currency*	*6.3*	*Total amount of fees.*
CertificateOfMailing	*Currency*	*6.3*	*Certificate of Mailing fee. For future use.*
CertifiedMail	*Currency*	*6.3*	*Certified Mail fee.*

Element	Type	Size or Values	Description
CollectOnDelivery	*Currency*	*6.3*	*Collect on Delivery (COD) fee.*
DeliveryConfirmation	*Currency*	*6.3*	*Delivery Confirmation fee.*
ElectronicReturnReceipt	*Currency*	*6.3*	*Electronic Return Receipt fee.*
InsuredMail	*Currency*	*6.3*	*Insured Mail fee.*
RegisteredMail	*Currency*	*6.3*	*Registered Mail fee.* *For future use.*
RestrictedDelivery	*Currency*	*6.3*	*Restricted Delivery fee.*
ReturnReceipt	*Currency*	*6.3*	*Return Receipt fee.*
ReturnReceiptForMerchandise	*Currency*	*6.3*	*Return Receipt for Merchandise fee.* *For future use.*
SignatureConfirmation	*Currency*	*6.3*	*Signature Confirmation fee.*
SpecialHandling	*Currency*	*6.3*	*Special Handling fee.* *For future use.*
AdultSignature	*Currency*	*6.3*	*Adult Signature fee.*
AdultSignatureRestrictedDelivery	*Currency*	*6.3*	*Adult Signature Restricted Delivery fee.*
LiveAnimalSurcharge	*Currency*	*6.3*	*Live Animal Surcharge fee.*

The following listing shows an example of the PostageRatesResponse XML.

Listing 9-2: PostageRatesResponse XML

```
<PostageRatesResponse>
  <Status>0</Status>
  <PostagePrice TotalAmount="4.53">
    <Postage TotalAmount="2.58">
      <MailService>First-Class Mail</MailService>
      <Zone>2</Zone>
      <IntraNDC>true</IntraNDC>
      <Pricing>Retail</Pricing>
    </Postage>
    <Fees TotalAmount="1.95">
      <CertificateOfMailing>0</CertificateOfMailing>
      <CertifiedMail>0</CertifiedMail>
      <CollectOnDelivery>0</CollectOnDelivery>
      <DeliveryConfirmation>0</DeliveryConfirmation>
      <ElectronicReturnReceipt>0</ElectronicReturnReceipt>
      <InsuredMail>0</InsuredMail>
      <RegisteredMail>0</RegisteredMail>
      <RestrictedDelivery>0</RestrictedDelivery>
      <ReturnReceipt>0</ReturnReceipt>
      <ReturnReceiptForMerchandise>0</ReturnReceiptForMerchandise>
      <SignatureConfirmation>1.95</SignatureConfirmation>
      <SpecialHandling>0</SpecialHandling>
    </Fees>
  </PostagePrice>
  <PostagePrice TotalAmount="7.39">
    <Postage TotalAmount="5.44">
      <MailService>Priority Mail</MailService>
      <Zone>2</Zone>
      <IntraNDC>true</IntraNDC>
```

```
      <Pricing>CommercialBase</Pricing>
    </Postage>
    <Fees TotalAmount="1.95">
      <CertificateOfMailing>0</CertificateOfMailing>
      <CertifiedMail>0</CertifiedMail>
      <CollectOnDelivery>0</CollectOnDelivery>
      <DeliveryConfirmation>0</DeliveryConfirmation>
      <ElectronicReturnReceipt>0</ElectronicReturnReceipt>
      <InsuredMail>0</InsuredMail>
      <RegisteredMail>0</RegisteredMail>
      <RestrictedDelivery>0</RestrictedDelivery>
      <ReturnReceipt>0</ReturnReceipt>
      <ReturnReceiptForMerchandise>0</ReturnReceiptForMerchandise>
      <SignatureConfirmation>1.95</SignatureConfirmation>
      <SpecialHandling>0</SpecialHandling>
    </Fees>
  </PostagePrice>
  <PostagePrice TotalAmount="24.97">
    <Postage TotalAmount="24.97">
      <MailService>Express Mail</MailService>
      <Zone>2</Zone>
      <IntraNDC>true</IntraNDC>
      <Pricing>CommercialBase</Pricing>
    </Postage>
    <Fees TotalAmount="0">
      <CertificateOfMailing>0</CertificateOfMailing>
      <CertifiedMail>0</CertifiedMail>
      <CollectOnDelivery>0</CollectOnDelivery>
      <DeliveryConfirmation>0</DeliveryConfirmation>
      <ElectronicReturnReceipt>0</ElectronicReturnReceipt>
      <InsuredMail>0</InsuredMail>
      <RegisteredMail>0</RegisteredMail>
      <RestrictedDelivery>0</RestrictedDelivery>
      <ReturnReceipt>0</ReturnReceipt>
      <ReturnReceiptForMerchandise>0</ReturnReceiptForMerchandise>
      <SignatureConfirmation>0</SignatureConfirmation>
      <SpecialHandling>0</SpecialHandling>
    </Fees>
  </PostagePrice>
  <PostagePrice TotalAmount="4.21">
    <Postage TotalAmount="2.26">
      <MailService>Library Mail</MailService>
      <Zone>2</Zone>
      <IntraNDC>true</IntraNDC>
      <Pricing>Retail</Pricing>
    </Postage>
    <Fees TotalAmount="1.95">
      <CertificateOfMailing>0</CertificateOfMailing>
      <CertifiedMail>0</CertifiedMail>
      <CollectOnDelivery>0</CollectOnDelivery>
      <DeliveryConfirmation>0</DeliveryConfirmation>
      <ElectronicReturnReceipt>0</ElectronicReturnReceipt>
      <InsuredMail>0</InsuredMail>
      <RegisteredMail>0</RegisteredMail>
      <RestrictedDelivery>0</RestrictedDelivery>
      <ReturnReceipt>0</ReturnReceipt>
      <ReturnReceiptForMerchandise>0</ReturnReceiptForMerchandise>
```

```
      <SignatureConfirmation>1.95</SignatureConfirmation>
      <SpecialHandling>0</SpecialHandling>
    </Fees>
  </PostagePrice>
  <PostagePrice TotalAmount="4.33">
    <Postage TotalAmount="2.38">
      <MailService>Media Mail</MailService>
      <Zone>2</Zone>
      <IntraNDC>true</IntraNDC>
      <Pricing>Retail</Pricing>
    </Postage>
    <Fees TotalAmount="1.95">
      <CertificateOfMailing>0</CertificateOfMailing>
      <CertifiedMail>0</CertifiedMail>
      <CollectOnDelivery>0</CollectOnDelivery>
      <DeliveryConfirmation>0</DeliveryConfirmation>
      <ElectronicReturnReceipt>0</ElectronicReturnReceipt>
      <InsuredMail>0</InsuredMail>
      <RegisteredMail>0</RegisteredMail>
      <RestrictedDelivery>0</RestrictedDelivery>
      <ReturnReceipt>0</ReturnReceipt>
      <ReturnReceiptForMerchandise>0</ReturnReceiptForMerchandise>
      <SignatureConfirmation>1.95</SignatureConfirmation>
      <SpecialHandling>0</SpecialHandling>
    </Fees>
  </PostagePrice>
  <PostagePrice TotalAmount="6.85">
    <Postage TotalAmount="4.9">
      <MailService>StandardPost</MailService>
      <Zone>2</Zone>
      <IntraNDC>true</IntraNDC>
      <Pricing>Retail</Pricing>
    </Postage>
    <Fees TotalAmount="1.95">
      <CertificateOfMailing>0</CertificateOfMailing>
      <CertifiedMail>0</CertifiedMail>
      <CollectOnDelivery>0</CollectOnDelivery>
      <DeliveryConfirmation>0</DeliveryConfirmation>
      <ElectronicReturnReceipt>0</ElectronicReturnReceipt>
      <InsuredMail>0</InsuredMail>
      <RegisteredMail>0</RegisteredMail>
      <RestrictedDelivery>0</RestrictedDelivery>
      <ReturnReceipt>0</ReturnReceipt>
      <ReturnReceiptForMerchandise>0</ReturnReceiptForMerchandise>
      <SignatureConfirmation>1.95</SignatureConfirmation>
      <SpecialHandling>0</SpecialHandling>
    </Fees>
  </PostagePrice>
</PostageRatesResponse>
```

10. GETTING ACCOUNT STATUS API

AccountStatusRequest

To get the status of an account, use the GetAccountStatus or the GetAccountStatusXML Web method of the Endicia Label Server Web Service. For accounts with an Active status, the response contains other useful information, such as the remaining postage balance in the account.

The following is a sample HTTP POST request for GetAccountStatusXML which requires an AccountStatusRequest XML string as input. Replace the **placeholders** with actual values:

```
POST /LabelService/EwsLabelService.asmx/GetAccountStatusXML HTTP/1.1
Host: www.envmgr.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

accountStatusRequestXML=<AccountStatusRequest> ... </AccountStatusRequest>
```

The response contains an AccountStatusResponse XML string:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<AccountStatusResponse> ... </AccountStatusResponse>
```

AccountStatusRequest XML Syntax

The following table describes the XML tags for AccountStatusRequest.

Important notes about all elements:

- Omitting any value causes the Endicia Label Server Web Service to use the default value for that element.
- *Italic* items are optional.

Table 10-1: AccountStatusRequest XML Elements

Element	Type	Size or Values	Description
AccountStatusRequest	**Root**	**-**	**The root node for the XML request.**
RequesterID	Text	50	Requester ID (also called Partner ID) is used to uniquely identify the system making the request. This ID is assigned by Endicia.
RequestID	Text	50	Request ID to uniquely identify this Account Status request. This will be returned in response.
CertifiedIntermediary	Node		Certified Intermediary (CI) account authentication information.
AccountID	Numeric	6	Account ID for the Endicia postage account.
PassPhrase	Text	64	Pass Phrase for the Endicia postage account.

The following listing shows an example of the AccountStatusRequest XML.

Listing 10-1: AccountStatusRequest XML

```
<AccountStatusRequest>
  <RequesterID>abcd</RequesterID>
  <RequestID>GAS12345</RequestID>
  <CertifiedIntermediary>
   <AccountID>123456</AccountID>
   <PassPhrase>samplePassPhrase</PassPhrase>
  </CertifiedIntermediary>
</AccountStatusRequest>
```

AccountStatusResponse XML Syntax

The following table describes the XML tags for AccountStatusResponse.

Table 10-2: AccountStatusResponse XML Elements

Element	Type	Size or Values	Description
AccountStatusResponse	**Root**	**-**	**The root node for the XML response.**
Status	Numeric	 0 Error code	Indicates if the account status request succeeded or failed: Success Failed. The response contains an ErrorMessage element describing the error.
ErrorMessage	*Text*	*150*	*Error message. See Status.*
RequesterID	*Text*	*50*	*Return of Requester ID based on input value.*
RequestID	*Text*	*50*	*Return of Request ID based on input value.*

Element	Type	Size or Values	Description
CertifiedIntermediary	*Node*		*Certified Intermediary (CI) account status.* *This node will only be present for an account with an Active status.*
AccountID	*Numeric*	*6*	*Account ID for the Endicia postage account.*
SerialNumber	*Numeric*	*12*	*The Serial Number of the last item created on the account.*
PostageBalance	*Currency*	*6.3*	*Remaining postage balance in dollars and cents (mils).*
AscendingBalance	*Currency*	*6.3*	*Total amount of postage printed in dollars and cents (mils).*
AccountStatus	*Text*	*A*	*Account Status flag:* *Active*
DeviceID	*Text*	*12*	*The account's 12 character Device ID (that appears in the indicium).*

The following listing shows an example of an AccountStatusResponse XML.

Listing 10-2: AccountStatusResponse XML

```
<AccountStatusResponse>
  <Status>0</Status>
  <RequesterID>abcd</RequesterID>
  <RequestID>GAS12345</RequestID>
  <CertifiedIntermediary>
   <AccountID>123456</AccountID>
   <SerialNumber>789</SerialNumber>
   <PostageBalance>76.55</PostageBalance>
   <AscendingBalance>123.45</AscendingBalance>
   <AccountStatus>A</AccountStatus>
   <DeviceID>071V00123456</DeviceID>
  </CertifiedIntermediary>
</AccountStatusResponse>
```

11. CREATING AND UPDATING SHARED PES USER CONTACT INFORMATION API

Background:

Postage Evidencing Systems (PES) are secure postage metering systems that generate indicium imprinted on or affixed to a mailpiece to evidence prepayment of postage. Recent trends in online business models have created systems that can support multiple users at multiple locations and hence, the concept of sharing a single PES.

Currently, with a shared PES implementation, the USPS is unable to accurately associate a mailpiece to the mailer. This hinders inspection service efforts and is inconsistent with USPS processes.

The USPS has mandated that all PC Postage vendors must implement a solution that tracks mailer contact information of all users of a shared PES, so that the known mailer status is maintained at all times.

For more information, specifically on adequate disclosure, compliance, denial and revocation of authorization to use, and other topics, see the USPS Domestic Mail Manual (DMM) section 604.4.2.

If you are developing software for use by multiple end-users who share one or multiple DYMO Endicia accounts, the USPS identifies you as a Shared PES Customer. Hence, the USPS requires that you send them the contact information of all your end-users prior to the users printing shipping labels. After the Shared PES User has been created in our system, the user must specify the <PartnerCustomerID> element when requesting a shipping label. If a valid <PartnerCustomerID> value is not specified, the label request will fail.

The following section provides details on an automated Web Service that allows you to create Shared PES User accounts more efficiently. You can collect all the needed contact information of your end-users and just submit it through this API. In return, you will receive an acknowledgement message.

SubmitUserContactInfo

Use the SubmitUserContactInfo or SubmitUserContactInfoXML Web method of the Endicia Label Server Web Service to send contact information of a new Shared PES User or update contact information of an existing Shared PES User.

The following is a sample HTTP POST request for SubmitUserContactInfoXML, which requires a SubmitUserContactInfo XML string as input. The **placeholders** shown need to be replaced with actual values:

```
POST /LabelService/SharedPESService.asmx/SubmitUserContactInfoXML HTTP/1.1
Host: www.envmgr.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

SubmitUserContactInfoXML=<UserContactInfoRequest> ...
</UserContactInfoRequest>
```

The response contains a SubmitUserContactInfoResponse XML string:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<UserContactInfoResponse> ... </UserContactInfoResponse>
```

SubmitUserContactInfoRequest XML Syntax

The following table describes the XML tags for SubmitUserContactInfoRequest.

Important notes about all elements:

- All contact information for a Shared PES User is required.
- The USPS requires that addresses of Shared PES Users be actual locations rather than P.O. Boxes™ and General Delivery addresses.
- *Italic* items are optional.

Table 11-1: SubmitUserContactInfoRequest XML Elements

Element	Type	Size or Values	Description
UserContactInfoRequest	**Root**	**-**	**The root node for the XML request.**
Test	*Attribute*	*YES* *NO*	*Test Request* *Live Request (Default)*
RequestID	Text	50	Request ID to uniquely identify this request. This will be returned in the response.
AccountID	Numeric	6	Account ID for the Endicia postage account associated with the Shared PES User. The Test Server does not authenticate the AccountID. Any 6-digit value is valid.
PassPhrase	Text	64	Pass Phrase for the Endicia postage account. The Test Server does not authenticate the PassPhrase. Any text value of 1 to 64 characters is valid.
PartnerCustomerID	Text	120	Custom-defined, unique Shared PES User ID. If PartnerCustomerID already exists, then the contact information will be updated.
Name	Text	50	Shared PES User Name; First name and last name, separated by a single space ‘ ‘.

Element	Type	Size or Values	Description
EmailAddress	Text	120	Shared PES User's Email Address
PhoneNumber	Number	10	Shared PES User's Phone Number; 10 digits required (including area code) with no punctuation. Use format 2125551212
PhysicalAddressLine1	Text	50	Shared PES User's Address
PhysicalAddressLine2	*Text*	*50*	*Additional Address Line*
PhysicalCity	Text	28	Shared PES User's City
PhysicalState	Text	2	Shared PES User's State
PhysicalZipCode	Number	5 or 9	Shared PES User's ZIP or ZIP+4; No punctuation or spaces.

XML Request Code Sample

The following listing shows an example of the SubmitUserContactInfoRequestXML.

Listing 11-1: SubmitUserContactInfoRequest XML

```
<UserContactInfoRequest Test="YES" >
  <RequestID>1234</RequestID>
  <AccountID>123456</AccountID>
  <PassPhrase>samplePassPhrase</PassPhrase>
  <PartnerCustomerID>ABC123</PartnerCustomerID>
  <Name>John Smith</Name>
  <EmailAddress>JSmith@domain.com</EmailAddress>
  <PhoneNumber>1234567890</PhoneNumber>
  <PhysicalAddressLine1>123 Address Lane #10</PhysicalAddressLine1>
  <PhysicalCity>Palo Alto</PhysicalCity >
  <PhysicalState>CA</PhysicalState>
  <PhysicalZipCode>943061234</PhysicalZipCode>
</UserContactInfoRequest>
```

SubmitUserContactInfoResponse XML Syntax

The following table describes the XML tags for SubmitContactInfoResponse.

Table 11-2: SubmitContactInfoResponse XML Elements

Element	Type	Size or Values	Description
UserContactInfoResponse	**Root**	**-**	**The root node for the XML response.**
ErrorMessage	*Text*	*150*	*Error Message*

Element	Type	Size or Values	Description
Status	Numeric	0 Error Code	Indicates if the request was successful Failed. The response contains an ErrorMessage element describing the error.
RequestID	Text	50	Return of Request ID submitted in the request.

XML Response Code Sample

The following listing shows an example of the UserContactInfoResponse XML.

Listing 11-2: SubmitUserContactInfoResponse XML

```
<UserContactInfoResponse>
     <ErrorMessage></ErrorMessage>
     <Status>0</Status>
     <RequestID>1234</RequestID>
</UserContactInfoResponse>
```

SubmitUserContactInfo Error Codes

Error Code String	Error Message
12144 - InvalidPartnerCustomerID	PartnerCustomerID is missing or invalid

12. REQUESTING A REFUND API

RefundRequest

If you are developing software for use by multiple customers and thus will be maintaining multiple accounts, this chapter demonstrates an automated Web Service that allows your customers to submit refund requests more efficiently. You can collect all the needed information and just submit it through this API. In return, you will receive an acknowledgement message.

Important! The URL for this service is distinct and different than the Endicia Label Server Web Service URL. For more information, see The Endicia Label Server Web Service.

RefundRequest XML Syntax

The following table describes the XML tags for RefundRequest.

Important notes about all elements:

- Omitting any value causes the Refund Web Service to use the default value for that element.
- *Italic* items are optional.
- The RefundList Node can have many Sub Nodes to do a list of shipments that need refunding.
- As a rule a tracking number is the PICNumber for a shipment. The technical term will always be PICNumber, but it is commonly referred to as a tracking number.
- You do not need both the PICNumber and PieceID for a refund, only one or the other. They are both in the API to give a choice as to which to use.
- You only need to submit one PieceID or one PICNumber per shipment each time you want to request a refund.
- It is recommended that you submit no more than 100 requests in a single XML transaction. If you submit more than 100 requests in a single XML transaction, your request may time out.
- If you include your Pass Phrase in the request, you do not need to submit your Web Password.

Table 12-1: RefundRequest XML Elements

Element	Type	Size or Values	Description
RefundRequest	**Root**	**-**	**The root node for the XML request.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
PassPhrase	Text	64	Customer's Pass Phrase
Test	*Text*	*Y or N*	*Is this a test request?*
RefundList	**Node**	**-**	**Container Node For the Refund List**

Element	Type	Size or Values	Description
PICNumber	*Numeric*	*30*	*Package PIC Number (Tracking Number)*
CustomsID	*Text*	*25*	*Package Customs ID FirstClassMailInternational, PriorityMailInternational and ExpressMailInternational.* *For more information, see Refund Policy.*
PieceID	*Numeric*	*10*	*Package Piece ID*

The following listing shows an example of the RefundRequest XML for the Refund Request API.

Listing 12-1: RefundRequest XML

```
<?xml version="1.0" encoding="utf-8"?>
<RefundRequest>
        <AccountID></AccountID>
        <PassPhrase></PassPhrase>
        <Test></Test>
        <RefundList>
                <PICNumber></PICNumber>
                <PieceID></PieceID>
                <CustomsID></CustomsID>
        </RefundList>
</ RefundRequest >
```

RefundResponse XML Syntax

The following table describes the XML tags for RefundResponse.

Table 12-2: RefundResponse XML Elements (XMLInput Parameter)

Element	Type	Size or Values	Description
RefundResponse	**Root**	**-**	**The root node for the XML response.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
ErrorMsg	*Text*	*-*	*Account Login Error Message*
FormNumber	*Numeric*	*6*	*Form Number for Refunded Shipments (Only for those approved)*
Test	*Text*	*Y or N*	*Is this a test request?*
RefundList	**Node**	**-**	**Container Node For the Refund List**
PICNumber	*Numeric*	*30*	*Package PIC Number*

Element	Type	Size or Values	Description
IsApproved	*Text*		*YES or NO*
ErrorMsg	*Text*		*If approved or disapproved, this indicates why.*
PieceID	*Numeric*	*10*	*Package Piece ID*
IsApproved	*Boolean*		*YES or NO*
ErrorMsg	*Text*		*If approved or disapproved, this will say why*
CustomsID	*Text*	*25*	*Package Customs ID PriorityMailInternational and ExpressMailInternational only.*
IsApproved	*Boolean*		*YES or NO*
ErrorMsg	*Text*		*If approved or disapproved, this will say why.*

The following listing shows an example of the RefundResponse XML (XMLInput Parameter).

Listing 12-2: RefundResponse XML (XMLInput Parameter)

```
<?xml version="1.0" encoding="UTF-8"?>
<RefundResponse>
        <AccountID></AccountID>
        <ErrorMsg></ErrorMsg>
        <FormNumber></FormNumber>
        <Test></Test>
        <RefundList>
                <PICNumber>
                        <IsAppoved></IsApproved>
                        <ErrorMsg></ErrorMsg>
                </PICNumber>
                <PieceID>
                        <IsAppoved></IsApproved>
                        <ErrorMsg></ErrorMsg>
                </PieceID>
                <CustomsID>
                        <IsApproved></IsApproved>
                        <ErrorMsg></ErrorMsg>
                </CustomsID>
        </RefundList>
</RefundResponse>
```

Using form submission: To use this API with a form submission, you must declare both the method and the XML's parameter name in the submission.

Example:

```
<form name="RefundRequest"
      method="POST"
      action="URL TO API&method=RefundRequest">
```

```
      <input type=”hidden” name=”XMLInput” value=”XML (Formatted as above)”>
</form>
```

OR (through cURL in PHP):

```
$xml = XML (Formatted as above);
$curl_handle = curl_init ();

curl_setopt ($curl_handle, CURLOPT_URL, ‘URL TO API’);
curl_setopt ($curl_handle, CURLOPT_FOLLOWLOCATION, 1);
curl_setopt ($curl_handle, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($curl_handle, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt ($curl_handle, CURLOPT_POST, 1);
curl_setopt ($curl_handle, CURLOPT_POSTFIELDS,
‘&method=RefundRequest&XMLInput=' . $xml);
$curl_result = curl_exec ($curl_handle) or die ("There has been a CURL_EXEC
error");
curl_close ($curl_handle);
echo $curl_result;
```

Example using SOAP submission (with PHP)

```
require_once('nusoap.php');
$soapclient = new soapclient(‘URL TO API’);
$xml = XML (Formatted as above);

$params = array(
        'XMLInput' => $xml
                );
$soapclient->call('RefundRequest',$params);
echo $soapclient->response;
```

RefundRequest Error Codes

Error Code String	Error Message
Login Failed	Given in the root node, occurs if the account number or password is incorrect.
Denied - Must be within 10 days.	Means that the refund was not requested within 10 days of the postage purchase. For electronic refunds this must always be within that 10-day period. (USPS rule)
Denied - Endicia Printable Postage is not eligible for refund.	You should never get this error, it is added in for completeness. We have another product Endicia Printable Postage that cannot ever be refunded.
Denied – Invalid	Given if the ID sent was empty.
Denied - Refund already submitted	Given if there already has been a refund request on this item.
Denied - Not Found	Given if there isn't any transaction in our system with that particular ID.
Approved - Less than 10 days.	All is well, Your refund has been submitted and approved.

Request Refunds Through Your Endicia.com Account

The following Steps describe how to request refunds through your endicia.com account.

Steps

1. Log in to your Endicia.com account using your Account Number and Internet Password.
2. From the My Account tab, select **Tools > Request a Postage Refund**.
3. Enter the criteria for your search, such as destination ZIP Code and date, tracking number, or date range.
4. Click **Go**.
5. In the left column, select the check boxes for the tracking numbers that you are requesting and click **Select Refund Request**.
6. Complete the process by clicking **Preview Refund Request** and then click **Submit Refund Request**.

View the Status of Previously Requested Refunds

The following Steps describe how to view the status of previously requested refunds.

Steps

1. Log in to your Endicia.com account using your Account Number and Internet Password.
2. From the My Account tab, select Reports > Refund Request History.
3. To see the individual pieces included in each request, click the **Form ID**.

Use the URL Method to Request a Refund

This is appropriate for creating one refund through a Web browser that the Partner would either integrate into their site using a frame or open to a new window to our site. It is, however, a manual process. https://www.endicia.com/ELS/RefundStart.cfm?accountnumber=xxxxxx&webpassword=xxxxxx(&postageidentifier)=xxxxxxxxxxxx where:

- &accountnumber is the users account number
- &webpassword is the user's Web password and
- (&postageidentifier) is *one* of three ways to identify the shipment:
    - &transactionid=91xxxxxxxxxxxxxx (picnumber)
    - &pieceid=xxxxxx (piecenumber)
    - &customsid=xxxxxxxxxxx (the customs ID number)
      Note: Use &customsid only for international shipments, where the customs ID number has been stored for tracking.

13. REQUESTING PACKAGE STATUS API

StatusRequest

If you are developing software for use by multiple customers and thus will be maintaining multiple accounts, this chapter demonstrates an automated Web Service that allows your customers to submit status requests more efficiently. You can collect all the needed information and just submit it through this API. In return, you will get an acknowledgement message back.

Important! The URL for this service is distinct and different than the Endicia Label Server Web Service URL. For more information, see page 20.

StatusRequest XML Syntax

The following table describes the XML tags for StatusRequest.

Important notes about all elements:

- Omitting any value will cause the Status Web Service to use the default value for that element.
- *Italic* items are optional.
- The StatusList Node can have many sub-nodes to do a list of shipments that need searching.
- As a rule a tracking number is the PICNumber for a shipment. The technical term will always be PICNumber, but it is commonly referred to as a tracking number.
- You do not need both the PICNumber and PieceID for a status request, just one or the other. They are both in the API to give a choice as to which to use.
- You only need to submit one PieceID or one PICNumber per shipment each time you want to request a status.
- The API is limited to 100 items per request.
- If you include your Pass Phrase in the request, you do not need to submit your Web password.
- The FullStatus element can be used to return a complete list of statuses for the requested packages.

Table 13-1: StatusRequest XML Elements (XMLInput Parameter)

Element	Type	Size or Values	Description
StatusRequest	**Root**	**-**	**The root node for the XML request.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
Test	*Text*	*Y or N*	*Is this a test request?*
PassPhrase	Text	64	Customer's Pass Phrase
TransactionID	*Numeric*		*Unique ID for transaction—corresponds to the piece count for the account used.*
ReferenceID	*Text*	*50*	*Returns the StatusList for the most recent 100 packages matching the specified ReferenceID and includes ReferenceID in the response.*
FullStatus	*Text*	*Yes* *No*	*Returns complete status history for the shipment.* *Returns the most recent status of the shipment.*
StatusList	**Node**	**-**	**Container Node For the Status List—Limited to any mixture of 100 of the following node types: PICNumber, CustomsID, and PieceID.**
PICNumber	*Numeric*	*30*	*Package PIC Number (Tracking Number)*
CustomsID	*Text*	*25*	*Package Customs ID*
PieceID	*Numeric*	*10*	*Package Piece ID*

The following listing shows a syntax example of the StatusRequest XML for the Refund Request API.

Listing 13-1: StatusRequest XML

```
<?xml version="1.0" encoding="utf-8"?>
<StatusRequest>
        <AccountID></AccountID>
        <Test></Test>
        <PassPhrase></PassPhrase>
        <FullStatus></FullStatus>
        <StatusList>
                <PICNumber></PICNumber>
                <PieceID></PieceID>
```

```
                <CustomsID></CustomsID>
        </StatusList>
</StatusRequest>
```

StatusResponse XML Syntax

The following table describes the XML tags for StatusResponse.

Table 13-2: StatusResponse XML Elements

Element	Type	Size or Values	Description
StatusResponse	**Root**	**-**	**The root node for the XML response.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
ErrorMsg	*Text*	*-*	*Account Login Error Message*
Test	*Text*	*Y or N*	*Is this a test request?*
StatusList	**Node**	**-**	**Container Node For the Status List**
PICNumber	*Numeric*	*30*	*Package PIC Number*
Status	*Text*		*Most recent status of the Shipment*
StatusBreakdown	***Node***	***-***	***Container Node for the Status Breakdown List***
*Status_*	*Text*		*Complete status history for the shipment, beginning with the most recent status* **Note:** *Each Status element is named with an underscore '_' followed by an ascending number beginning with 1.*
StatusCode	*Text*		*USPS Status Code of the Shipment*
PieceID	*Numeric*	*10*	*Package Piece ID*
Status	*Text*		*Most recent status of the Shipment*
StatusCode	*Text*	*1*	*USPS Status Code of the Shipment*
CustomsID	*Text*	*25*	*Package Customs ID*
Status	*Text*		*Most recent status of the Shipment*
StatusCode	*Text*	*1*	*USPS Status Code of the Shipment*

The following listing shows an example of the StatusResponse XML.

Listing 13-2: StatusResponse XML

```
<?xml version="1.0" encoding="UTF-8"?>
<StatusResponse>
        <AccountID></AccountID>
        <ErrorMsg></ErrorMsg>
        <Test></Test>
        <StatusList>
                <PICNumber>
                        <Status></Status>
                                <StatusBreakdown>
                                        <Status_1></Status_1>
                                        <Status_2></Status_2>
                                                :
                                                :
                                        <Status_n></Status_n>
                                </StatusBreakdown>
                        <StatusCode></StatusCode>
                </PICNumber>
                <PieceID>
                        <Status></Status>
                        <StatusCode></StatusCode>
                </PieceID>
                <CustomsID>
                        <Status></Status>
                        <StatusCode></StatusCode>
                </CustomsID>
        </StatusList>
</StatusResponse>
```

Using form submission: To use this API with a form submission, you must declare both the method and the XML's parameter name in the submission.

Example:

```
<form name="StatusRequest"
      method="POST"
      action="URL TO API&method=StatusRequest">
      <input type="hidden" name="XMLInput" value="XML (Formatted as above)">
</form>
```

OR (through cURL in PHP):

```
$xml = XML (Formatted as above);
$curl_handle = curl_init ();

curl_setopt ($curl_handle, CURLOPT_URL, 'URL TO API');
curl_setopt ($curl_handle, CURLOPT_FOLLOWLOCATION, 1);
curl_setopt ($curl_handle, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($curl_handle, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt ($curl_handle, CURLOPT_POST, 1);
curl_setopt ($curl_handle, CURLOPT_POSTFIELDS,
'&method=StatusRequest&XMLInput=' . $xml);
$curl_result = curl_exec ($curl_handle) or die ("There has been a CURL_EXEC
error");
curl_close ($curl_handle);
echo $curl_result;
```

Example using SOAP submission (with PHP)

```
require_once('nusoap.php');
$soapclient = new soapclient('URL TO API');
$xml = XML (Formatted as above);

$params = array(
        'XMLInput' => $xml
                );$soapclient->call('StatusRequest',$params);

echo $soapclient->response;
```

StatusRequest Error Codes

Error Code String	Error Message
Login Failed	Given in the root node, occurs if the account number or password is incorrect.
Not Found	Happens if the item does not belong to the account making the request, or the item is not in our system at all.
Found – No Status	Happens if the USPS has not updated the status of the item, but it is found in our system.

StatusCode Definitions

Note: These are not exact definitions.

Status Code String	Status Message
-1	Not Found
0	Found – No Status
A	Logged at USPS
I	Scanned in Route
D	Delivered
L	Notice Left for Recipient
F	Forwarded to New Address
R	Returned to Sender
U	Undeliverable
N	New Item
X	Other

Obtain Package Status Requests Through Your Endicia.com Account

The following Steps describe how to obtain package status information through your endicia.com account.

Steps

1. Log in to your Endicia.com account using your Account Number and Internet Password.
2. From the My Account tab, select **Reports > Account Transactions**.
   The following search criteria are available:
   - Month
   - Period

   You can filter the items included in the response by selecting from the two Show lists, and order the transactions in ascending or descending order according to several criteria.
3. To view the data in a spreadsheet instead of on the page, select the **Export to Spreadsheet** check box.
   Tip! The search results are limited to the most recent 1,000 transactions. If you require more transactions, you can tighten the date range and merge multiple spreadsheets.
4. To see a graphical representation of your delivery statistics, select **Reports > Delivery Statistics** from the My Account tab.

14. REQUESTING TRANSACTION LISTS API

GetTransactionsListing

If you are developing software for use by multiple customers and thus will be maintaining multiple accounts, this chapter demonstrates an automated Web Service that allows your customers to get transaction listings more efficiently. You can collect all the needed information and just submit it through this API. In return, you will get an acknowledgement message back.

Important! The URL for this service is distinct and different than the Endicia Label Server Web Service URL. For more information, see The Endicia Label Server Web Service.

GetTransactionsListing XML Syntax

The following table describes the XML tags for GetTransactionsListing.

Important notes about all elements:

- Omitting any value will cause the Get Transactions Listing Web Service to use the default value for that element.
- *Italic* items are optional.
- The TrackingList Node can have many Sub Nodes to do a list of shipments that need searching.
- As a rule a tracking number is the PICNumber for a shipment. The technical term will always be PICNumber, but it is commonly referred to as a tracking number.
- You do not need both the PICNumber and PieceID for a status request, just one or the other. They are both in the API to give a choice as to which to use.
- You only need to submit one PieceID or one PICNumber per shipment each time you want to request a status. GetTransactionListing returns a maximum of 50 results per request.
- Date ranges are limited to 7 days.
- The maximum number of results is 50.
- If used with no optional inputs, it will return the latest 50 transactions with the latest one being first sorted by time.
- If you include your Pass Phrase in the request, you do not need to submit your Web Password.

Table 14-1: GetTransactionsListing XML Elements (XMLInput Parameter)

Element	Type	Size or Values	Description
TransactionsListingRequest	**Root**	**-**	**The root node for the XML request.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
PassPhrase	Text	64	Customer's Pass Phrase
Test	*Text*	*Y or N*	*Is this a test request?*
TransactionType	*Text*	*Prints* *Purchases* *All*	*Returns print transactions only (default).* *Returns purchase transactions only* *Returns both print and purchase transactions.*
CostCenter	*Text*	*All* *XXXX*	*Includes CostCenter in response.* *Returns transactions with specified CostCenter, and includes CostCenter in response.*
ReferenceID	*Text*	*50*	*Returns the Transaction Listing for the most recent 50 packages matching the specified ReferenceID and includes ReferenceID in the response.*
FromAddress	*Text*	*Y or N*	*Do you want the from address information returned?*
StartDate	*Date*	*mm/dd/yyyy*	*Start Date of the Query*
StartTime	*Time*	*hh:mm:ss*	*Start Time of the Query*
EndDate	*Date*	*mm/dd/yyyy*	*End Date of the Query*

Element	Type	Size or Values	Description
EndTime	*Time*	*hh:mm:ss*	*End Time of the Query*
ToZipCode	*ZipCode*	*XXXXX*	*Destination ZIP Code To Search For*
Results	*Int*	*1 – 50*	*Maximum Results Returned (If more than 50 is used, it will default to 50)*
TrackingList	***Node***	***-***	***Container Node For the Tracking List***
PICNumber	*Numeric*	*30*	*Package PIC Number (Tracking Number)*
CustomsID	*Text*	*25*	*Package Customs ID*
PieceID	*Numeric*	*10*	*Package Piece ID*

The following listing shows a syntax example of the StatusRequest XML for the Refund Request API.

Listing 14-1: GetTransactionsListing XML

```
<?xml version="1.0" encoding="utf-8"?>
<TransactionsListingRequest>
        <AccountID></AccountID>
        <PassPhrase></PassPhrase>
        <Test></Test>
        <TransactionType></TransactionType>
        <StartDate></StartDate>
        <StartTime></StartTime>
        <EndDate></EndDate>
        <EndTime></EndTime>
        <Results></Results>
        <ToZipCode></ToZipCode>
        <FromAddress></FromAddress>
        <TrackingList>
                <PICNumber></PICNumber>
                <PieceID></PieceID>
                <CustomsID></CustomsID>
        </TrackingList>
</TransactionsListingRequest>
```

TransactionsListingResponse XML Syntax

The following table describes the XML tags for TransactionsListingResponse.

Table 14-2: GetTransactionsListing Response XML Elements

Element	Type	Size or Values	Description
TransactionsListingResponse	**Root**	**-**	**The root node for the XML response.**
TransactionType	Text		Transaction Type – Print or Purchase
AccountID	Numeric	6	Account ID for the Endicia postage account.
ErrorMsg	*Text*	-	*Error Message*
Test	*Text*	*Y or N*	*Is this a test request?*
TransactionResults	**Node**	**-**	**Container Node for the Transaction List**
Transaction	Node	-	Container Node for the Transaction
PieceID	Numeric		Piece Number of the shipment
Amount	Numeric		Value of the postage
Weight	Text		Weight of the shipment
TransactionDateTime	DateTime	mm/dd/yyyy hh:mm:ss	Date and time the shipment was created

Element	Type	Size or Values	Description
PostmarkDate	Date	mm/dd/yyyy	Date of the postmark on the shipment
MailClass	Text		List of the shipment definition delimited by ';' (For example: "First-Class; Delivery Confirmation")
CustomsID	Text		Customs ID of the shipment
PICNumber	*Text*		*Tracking Number of the shipment*
ToName	Text		Name of the person receiving the shipment
ToCompany	Text		Company name of the receiving party
ToAddress	Text		Address of the receiving party
ToCity	Text		City of the receiving party
ToState	Text		State of the receiving party
ToZipCode	Text	XXXXX-XXXX	ZIP Code of the receiving party
ToPhone	Text	XXXXXXXXXX	Phone number of the receiving party
ToEmail	Text		Email address of the receiving party
ToCountry	Text		Country of the receiving party
FromName	*Text*		*Name of the person sending the shipment. shown if FromAddress is Y.*
FromCompany	*Text*		*Company name of the sending party. Shown if FromAddress is Y.*
FromAddress	*Text*		*Address of the sending party. Shown if FromAddress is Y.*
FromCity	*Text*		*City of the sending party. Shown if FromAddress is Y.*
FromState	*Text*		*State of the sending party. Shown if FromAddress is Y.*
FromZipCode	*Text*	*XXXXX-XXXX*	*ZIP Code of the sending party. Shown if FromAddress is Y.*
FromPhone	*Text*	*XXXXXXXXXX*	*Phone Number of the sending party. Shown if FromAddress is Y.*
FromEmail	*Text*		*Email address of the sending party. Shown if FromAddress is Y.*
DeliveryDateTime	*DateTime*	*mm/dd/yyyy hh:mm:ss*	*Date and time of the delivery*

Element	Type	Size or Values	Description
StatusCode	Text		USPS Status Code of the shipment
Status	Text		USPS status of the shipment
RefundCode	*Char*	*1*	*Refund Code of the shipment*
RefundStatus	*Text*		*Refund definition of the shipment*

The following listing shows an example of the GetTransactionsListing Response XML.

Listing 14-2: GetTransactionsListing Response XML

```
<?xml version="1.0" encoding="UTF-8"?>
<TransactionsListingResponse>
        <AccountID></AccountID>
        <ErrorMsg></ErrorMsg>
        <Test></Test>
        <TransactionResults>
                <Transaction>
                   <TransactionType></TransactionType>
                   <PieceID></PieceID>
                   <Amount></Amount>
                   <Weight></Weight>
                   <TransactionDateTime></TransactionDateTime>
                   <PostmarkDate></PostmarkDate>
                   <MailClass></MailClass>
                   <CustomsID></CustomsID>
                   <PICNumber></PICNumber>
                   <ToName></ToName>
                   <ToCompany></ToCompany>
                   <ToAddress></ToAddress>
                   <ToCity></ToCity>
                   <ToState></ToState>
                   <ToZipCode></ToZipCode>
                   <ToPhone></ToPhone>
                   <ToEmail></ToEmail>
                   <ToCountry></ToCountry>
                   <FromCompany</FromCompany>
                   <FromName></FromName>
                   <FromAddress></FromAddress>
                   <FromCity></FromCity>
                   <FromState></FromState>
                   <FromZipCode></FromZipCode>
                   <FromPhone></FromPhone>
                   <FromEmail></FromEmail>
                   <DeliveryDateTime></DeliveryDateTime>
                   <StatusCode></StatusCode>
                   <Status></Status>
                   <RefundCode></RefundCode>
                   <RefundStatus></RefundStatus>
                </Transaction>
        </TransactionResults>
</TransactionsListingResponse>
```

GetTransactionsListing Response Error Codes

Error Code String	Error Message
Login Failed	Given in the root node, occurs if the account number or password is incorrect.
ERROR 104: Please limit requests to seven days	Please limit requests to seven days

GetTransactionsListing Response Definitions

See Status Code Definitions on page 103.

GetTransactionsListing Refund Code and Refund Status

Refund Code	Refund Status
X	Refund Submitted
Y	Refund Validated
R	Refund Rejected
A,C, S, U	Refunded
M	Manual Refund

View Transactions Listing Through Your Endicia.com Account

Alternatively, you can retrieve transactions through your endicia.com account using the same means as you would for status lists. See page 104.

15. USING THE SCAN FORM API

SCANRequest

SCAN associates all trackable items to a single form containing a barcode, which the USPS scans when your packages are picked up. When tracking numbers are uploaded into the postal system that evening, all your packages are marked "Shipment Accepted" on the USPS tracking system.

When your customer verifies the status of their package, instead of reading that the USPS was electronically notified of the item, your customer will see "Shipment Accepted" followed by the date, time, and location.

Pickup Service is not integrated with SCAN. Use the PickupRequest Web method to schedule a pickup.

> **Important!** The URL for this service is distinct and different than the Endicia Label Server Web Service URL. For more information, see The Endicia Label Server Web Service.
>
> **Important!** Be sure to submit all refund requests before running SCAN. Tracking numbers included in a SCAN Form **cannot be refunded**.

SCANRequest XML Syntax

The following table describes the XML tags for SCANRequest

Important notes about all elements:

Omitting any value will cause the SCAN Request Web Service to use the default value for that element.

- *Italic* items are optional.
- You can enter each tracking number in the SCANList Node individually. Or, you can use the CostCenter to automatically add all tracking numbers with the same CostCenter value, or add all eligible tracking numbers to the SCAN by omitting the `<SCANList>` node. In its simplest form, a valid SCAN Request could look like the following:

```
<SCANRequest>
   <ImageFormat></ImageFormat>
   <AccountID></AccountID>
   <PassPhrase></PassPhrase>
</SCANRequest>
```

- The tracking number is also the PICNumber for a shipment. The technical term will always be PICNumber, but it is commonly referred to as a tracking number.
- Choose PICNumber, CustomsID, or PieceID for each package. Each is included in the API so that you can choose your preference.
- All package labels must have the same date (TransactionDateTime) and originate from the same ZIP Code.
- There is no limit to the number of items that can be included in a SCAN Form.

Table 15-1: SCANRequest XML Elements

Element	Type	Size or Values	Description
SCANRequest	**Root**	**-**	**The root node for the XML**

Element	Type	Size or Values	Description
			request.
AccountID	Numeric	6	Account ID for the Endicia postage account.
PassPhrase	Text	64	Customer's Pass Phrase
FormType	Text	5630 (Default) 3152	USPS PS-5630 SCAN form USPS PS-3152 SCAN form (for Open and Distribute services)
Test	*Text*	*Y or N*	*Is this a test transaction?*
FromName	*Text*	*50*	*Name on the form (defaults to the name on the account)*
FromAddress	*Text*	*100*	*Address on the form (defaults to the address on the account)*
FromCity	*Text*	*50*	*City on the form (defaults to the city on the account)*
FromState	*Text*	*2*	*State abbreviation on the form (defaults to the state on the account)*
FromZipCode	*Text*	*5 or 10 digits xxxxx or xxxxx-xxxx*	*ZIP Code on the form (defaults to the ZIP Code on the account)*
SubmissionID	*Text*		*Submission ID of the form. Enter the SubmissionID of a previously generated SCAN Form to obtain a reprint.*
ImageFormat	*Text*	*PDF, GIF, PNG, or JPG*	*Format of the image returned in base64 format (GIF is default).*
CostCenter	*Text*	*15*	*Can be used instead of SCANList to auto-add all pieces with that grouping code.*
ReferenceID	*Text*	*50*	*Can be used instead of SCANList to automatically add all pieces with that ReferenceID.*
DPI	*Numeric*	*96, 150, or 300*	*Defaults to 96, but can be 96,150, or 300.*
SCANList	***Node***	**—**	***Container Node For the SCAN List.*** *Use no more than* ***one*** *option for each package.*
PICNumber	*Numeric*	*30*	*Package PIC or Tracking Number*
CustomsID	*Text*	*25*	*Package Customs ID*
PieceID	*Numeric*	*10*	*Package Piece ID*

The following listing shows a syntax example of the SCANRequest XML for the SCAN Request API.

Listing 15-1: SCANRequest XML

```
<?xml version="1.0" encoding="utf-8"?>
<SCANRequest>
```

```
        <AccountID></AccountID>
        <PassPhrase></PassPhrase>
        <Test></Test>
        <DPI></DPI>
        <SCANList>
                <PICNumber></PICNumber>
                <PieceID></PieceID>
                <CustomsID></CustomsID>
        </SCANList>
</SCANRequest>
```

SCANResponse XML Syntax

The following table describes the XML tags for SCANResponse.

Table 15-2: SCANResponse XML Elements (XMLInput Parameter)

Element	Type	Size or Values	Description
SCANResponse	**Root**	**-**	**The root node for the XML response.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
ErrorMsg	*Text*	-	*Error Message*
Test	*Text*	*Y or N*	*Is this a test transaction?*
SubmissionID	Numeric		Submission ID of the SCAN Form
ImageFormat	Text	GIF, PNG, or JPG	Image format of the SCAN Form
SCANForm	Base64 encoded image		SCANForm Image

The following listing shows an example of the SCANResponse XML.

Listing 15-2: SCANResponse XML

```
<?xml version="1.0" encoding="UTF-8"?>
<SCANResponse>
        <AccountID></AccountID>
        <ErrorMsg></ErrorMsg>
        <Test></Test>
        <SubmissionID></SubmissionID>
        <ImageFormat></ImageFormat>
        <DPI></DPI>
        <SCANForm></SCANForm>
</SCANResponse>
```

Using form submission: To use this API with a form submission you must declare both the method and the XML's parameter name in the submission.

Example:

```
<form name="SCANRequest"
      method="POST"
      action="URL TO API&method=SCANRequest">
      <input type="hidden" name="XMLInput" value="XML (Formatted as above)">
```

```
</form>
```

OR (through cURL in PHP):

```
$xml = XML (Formatted as above);
$curl_handle = curl_init ();

curl_setopt ($curl_handle, CURLOPT_URL, 'URL TO API');
curl_setopt ($curl_handle, CURLOPT_FOLLOWLOCATION, 1);
curl_setopt ($curl_handle, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($curl_handle, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt ($curl_handle, CURLOPT_POST, 1);
curl_setopt ($curl_handle, CURLOPT_POSTFIELDS,
'&method=SCANRequest&XMLInput=' . $xml);
$curl_result = curl_exec ($curl_handle) or die ("There has been a CURL_EXEC
error");
curl_close ($curl_handle);
echo $curl_result;
```

Example using SOAP submission (with PHP)

```
require_once('nusoap.php');

$soapclient = new soapclient('URL TO API');

$xml = XML (Formatted as above);

$params = array(
        'XMLInput' => $xml
                );

$soapclient->call('SCANRequest',$params);

echo $soapclient->response;
```

SCANRequest Error Messages

ErrorMsg String	Error Message
Login Failed	Given in the root node, occurs if the account number or password is incorrect.
XMLInput Error	XML is not formatted correctly.
Error 102: No shipments added to SCAN. Please check tracking numbers.	Either the shipments have already been added to another form, or they do not qualify for SCAN.
Error 105: SCAN Form Not Found	If the SCAN form cannot be found for that account.

Create a New SCAN Form Through Your Endicia Account

The following Steps describe how to create SCAN Forms through your endicia.com account.

Steps

1. Log in to your Endicia.com account using your Account Number and Internet Password.
2. From the My Account tab, select **Tools > Create a SCAN Form.**
3. Select the mailpieces to include in the request. Include any applicable filters.
4. Click **Continue SCAN Request**.

Create a SCAN Form History Report Through Your Endicia Account

The following Steps describe how to retrieve past SCAN Forms.

Steps

1. Log in to your Endicia.com account using your Account Number and Internet Password.
2. From the My Account tab, select **Tools > SCAN Form History.**
3. Select the ID for the date that you want.
4. Select the **SCAN Form** link above the list.
   You can reprint any form as well as select the FormID to see a full list of tracking numbers included in the SCAN.

16. REQUESTING A CARRIER PICKUP API'S

If you are developing software for use by multiple customers and thus will be maintaining multiple accounts, this chapter demonstrates an automated Web Service that will allow your customers to submit carrier pickup requests more efficiently. You can collect all the needed information and just submit it through this API. In return, you will receive an acknowledgement message.

The Requesting a Carrier Pickup API includes:

- Scheduling a Carrier Pickup API
- Canceling a Carrier Pickup Request API
- Changing a Carrier Pickup Request API
- Scheduling a Multiple Location Carrier Pickup Request API

Important! The URL for this service is distinct and different than the Endicia Label Server Web Service URL. For more information, see The Endicia Label Server Web Service.

Important notes about all elements:

- Omitting any value will cause the Web Service to use the default value for that element.
- *Italic* items are optional.
- The PickupList Node, if needed, can have many Sub Nodes to do a list of shipments that need to be added to the carrier pickup.
- As a rule a tracking number is the PICNumber for a shipment. The technical term will always be PICNumber, but it is commonly referred to as a tracking number.
- You do not need both the PICNumber, CustomsID and PieceID, just one of them. They are all in the API to give a choice as to which to use.
- You may submit as many shipments as you want per request.

Important notes about these API's:

- Express Mail International is counted as "Express".
- Priority Mail International is counted as "Priority".
- At least one Express or Priority mail shipment must be included in the pickup.
- A pickup for the next day must be scheduled prior to 2AM EST the day of the pickup.
- A pickup for the next day can be changed and cancelled prior to 2AM EST the day of the pickup.
- You must have the same address information and confirmation number to change or cancel a pickup.
- **You need special permission to use the Multiple Location Carrier Pickup API.** You must contact us to become approved.
- Locations for the Multiple Location Carrier Pickup API are grouped by the Cost Center tag. (LabelRequest XML)
- Address information for the Multiple Location Carrier Pickup defaults to first shipment found in that Cost Center
- If you include your Pass Phrase in the request, you do not need to submit your Web Password.

17. SCHEDULING A CARRIER PICKUP API

CarrierPickupRequest

Table 15-17-1: Carrier Pickup Request API

Element	Type	Size or Values	Description
CarrierPickupRequest	**Root**	**-**	**The root node for the XML request.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
PassPhrase	Text	64	Customer's Pass Phrase
Test	*Text*	*Y or N*	*Is this a test transaction?*
UseAddressOnFile	*Text*	*Y or N*	*Use your physical address we have on file (Default Y)*
FirstName	*Text*	*100*	*First name on the pickup (defaults to the first name on the account)*
LastName	*Text*	*100*	*Last name on the pickup (defaults to the last name on the account)*
CompanyName	*Text*	*100*	*Company name on the pickup (defaults to the company name on the account)*
SuiteOrApt	*Text*	*25*	*Suite or apartment number on the pickup (defaults to the one on file)*
Address	*Text*	*100*	*Address of the pickup (defaults to the one on file)*
City	*Text*	*100*	*City of the pickup (defaults to the one on file)*

Element	Type	Size or Values	Description
State	*Text*	*2*	*State abbreviation of the pickup (defaults to the one on file)*
ZIP5	*Numeric*	*5*	*Five digit ZIP Code of the pickup (defaults to the one on file)*
ZIP4	*Numeric*	*4*	*+4 ZIP Code for the pickup (defaults to the one on file)*
Phone	*Numeric*	*10*	*Phone number of the pickup (defaults to the one on file)*
Extension	*Numeric*	*5*	*Phone number extension for the pickup (defaults to the one on file)*
PackageLocation	Char	2	sd - Side Door kd - Knock on Door/Ring Bell mr - Mail Room of - Office rc - Reception im - In/At Mailbox ot - Other fd - Front Door bd - Back Door
SpecialInstructions	*Text*	*100*	*Required if "ot" chosen for PackageLocation. Special Instructions for pickup.*
PickupList	*Node*		*Shipments to be picked up.* *They can be referred to by PIC, Customs ID, or Piece ID. You only need one reference per shipment.*
PICNumber	*Numeric*	*30*	*Package PIC Number (Tracking Number)*
CustomsID	*Text*	*25*	*Package Customs ID*
PieceID	*Numeric*	*10*	*Package Piece ID*

Notes

1. **Automated Carrier Pickup.** Endicia has automated USPS Carrier Pickup. This feature can be used to schedule next-day pickup by the Postal Service for packages created by the Endicia Label Server (Express Mail, Priority Mail, and international shipments). More information is available upon request.

The following listing shows an example of the CarrierPickupRequest XML for the Carrier Pickup Request API.

Listing 17-1: CarrierPickupRequest XML

```
<?xml version="1.0" encoding="utf-8"?>
<CarrierPickupRequest>
        <AccountID></AccountID>
        <PassPhrase></PassPhrase>
        <Test></Test>
        <PackageLocation></PackageLocation>
        <PickupList>
                <PICNumber></PICNumber>
                <PieceID></PieceID>
                <CustomsID></CustomsID>
        </PickupList>
</CarrierPickupRequest>
```

CarrierPickupResponse XML Syntax

The following table describes the XML tags for CarrierPickupResponse.

Table 17-2: CarrierPickupResponse XML Elements (XMLInput Parameter)

Element	Type	Size or Values	Description
CarrierPickupRequestResponse	**Root**	**-**	**The root node for the XML response.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
ErrorMsg	*Text*	*-*	*Error Message*
Test	*Text*	*Y or N*	*Is this a test transaction?*
Response	**Container Node**		**Container Node for the Response**
Error	*Container Node*		*Container Node For the Error (will only show IF there is an error)*
Number	*Numeric*		*USPS Error Code*
Description	*Text*	*250*	*USPS Error Code Description*
DayOfWeek	Text	Pickup Day of week (that is, “Monday”)	Pickup day of week
Date	Date	xx/xx/xxxx	Date of pickup
CarrierRoute	Char	1	Pickup Route type (that is, “C” for City)
ConfirmationNumber	Text	100	Confirmation number of the pickup (Needed to change or cancel pickup)

The following listing shows an example of the CarrierPickupResponse XML.

Listing 17-2: CarrierPickupResponse XML

```
<?xml version="1.0" encoding="UTF-8"?>
<CarrierPickupResponse>
        <AccountID></AccountID>
        <ErrorMsg></ErrorMsg>
        <Test></Test>
        <Response>
                <DayOfWeek></DayOfWeek>
                <Date></Date>
                <CarrierRoute></CarrierRoute>
                <ConfirmationNumber></ConfirmationNumber>
        </Response>
</CarrierPickupResponse>
```

Using form submission: To use this API with a form submission you must declare both the method and the XML's parameter name in the submission.

Example:

```
<form name="CarrierPickupRequest"
      method="POST"
      action="URL TO API&method=CarrierPickupRequest">
      <input type="hidden" name="XMLInput" value="XML (Formatted as above)">
</form>
```

OR (through cURL in PHP):

```
$xml = XML (Formatted as above);
$curl_handle = curl_init ();

curl_setopt ($curl_handle, CURLOPT_URL, 'URL TO API');
curl_setopt ($curl_handle, CURLOPT_FOLLOWLOCATION, 1);
curl_setopt ($curl_handle, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($curl_handle, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt ($curl_handle, CURLOPT_POST, 1);
curl_setopt ($curl_handle, CURLOPT_POSTFIELDS,
'&method=CarrierPickupRequest&XMLInput=' . $xml);
$curl_result = curl_exec ($curl_handle) or die ("There has been a CURL_EXEC
error");
curl_close ($curl_handle);
echo $curl_result;
```

Example using SOAP submission (with PHP)

```
require_once('nusoap.php');

$soapclient = new soapclient('URL TO API');

$xml = XML (Formatted as above);

$params = array(
        'XMLInput' => $xml
                );

$soapclient->call('CarrierPickupRequest',$params);

echo $soapclient->response;
```

CarrierPickupRequest Error Codes

Error Code String	Error Message
Login Failed	Given in the root node, occurs if the account number or password is incorrect.
XMLInput Error	XML is not formatted correctly.
You need at least 1 Express or Priority package	You need at least one Priority or Express mail shipment.
Invalid Package Location	You must use: sd for Side Door kd for Knock on Door/Ring Bell mr for Mail Room of for Office rc for Reception im for In/At Mailbox ot for Other fd for Front Door bd for Back Door
USPS Error Codes	**No complete list is available.**

18. CANCELING A CARRIER PICKUP REQUEST API

CarrierPickupCancel

Table 18-1: CarrierPickupCancel XML Elements

Element	Type	Size or Values	Description
CarrierPickupCancel	**Root**	**-**	**The root node for the XML request.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
PassPhrase	Text	64	Customer's Pass Phrase
Test	*Text*	*Y or N*	*Is this a test transaction?*
UseAddressOnFile	*Text*	*Y or N*	*Use your physical address we have on file (Default Y)*
CompanyName	*Text*	*100*	*Company name on the pickup (defaults to the company name on the account)*
SuiteOrApt	*Text*	*25*	*Suite or apartment number on the pickup (defaults to the one on file)*
Address	*Text*	*100*	*Address of the pickup (defaults to the one on file)*
City	*Text*	*100*	*City of the pickup (defaults to the one on file)*
State	*Text*	*2*	*State abbreviation of the pickup (defaults to the one on file)*
ZIP5	*Numeric*	*5*	*Five digit ZIP Code of the pickup (defaults to the one on file)*

Element	Type	Size or Values	Description
ZIP4	*Numeric*	*4*	*+4 ZIP Code for the pickup (defaults to the one on file)*
ConfirmationNumber	Text	100	Confirmation number for the pickup

The following listing shows a syntax example of the CarrierPickupCancel XML for the Carrier Pickup Cancel API.

Listing 18-1: CarrierPickupCancel XML

```
<?xml version="1.0" encoding="utf-8"?>
<CarrierPickupCancel>
        <AccountID></AccountID>
        <PassPhrase></PassPhrase>
        <Test></Test>
        <ConfirmationNumber></ConfirmationNumber>
</CarrierPickupCancel>
```

CarrierPickupCancelResponse XML Syntax

The following table describes the XML tags for CarrierPickupCancelResponse.

Table 18-2: CarrierPickupCancelResponse XML Elements (XMLInput Parameter)

Element	Type	Size or Values	Description
CarrierPickupCancelResponse	**Root**	**-**	**The root node for the XML response.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
ErrorMsg	*Text*	*-*	*Error Message*
Test	*Text*	*Y or N*	*Is this a test transaction?*
Response	**Container Node**		**Container Node for the Response**
Error	*Container Node*		*Container Node For the Error (will only show IF there is an error)*
Number	*Numeric*		*USPS Error Code*
Description	*Text*	*250*	*USPS Error Code Description*
ConfirmationNumber	Text	100	Confirmation Number of the Pickup (Needed to change or cancel pickup)
Status	Text	250	Status of the pickup (that is, “Cancelled”)

The following listing shows an example of the CarrierPickupCancelResponse XML.

Listing 18-2: CarrierPickupCancelResponse XML

```
<?xml version="1.0" encoding="UTF-8"?>
<CarrierPickupResponse>
        <AccountID></AccountID>
        <ErrorMsg></ErrorMsg>
        <Test></Test>
        <Response>
                <DayOfWeek></DayOfWeek>
                <Date></Date>
                <CarrierRoute></CarrierRoute>
                <ConfirmationNumber></ConfirmationNumber>
        </Response>
</CarrierPickupResponse>
```

Using form submission: To use this API with a form submission you must declare both the method and the XML's parameter name in the submission.

Example:

```
<form name="CarrierPickupCancel"
      method="POST"
      action="URL TO API&method=CarrierPickupCancel">
      <input type="hidden" name="XMLInput" value="XML (Formatted as above)">
</form>
```

OR (through cURL in PHP):

```
$xml = XML (Formatted as above);
$curl_handle = curl_init ();

curl_setopt ($curl_handle, CURLOPT_URL, 'URL TO API');
curl_setopt ($curl_handle, CURLOPT_FOLLOWLOCATION, 1);
curl_setopt ($curl_handle, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($curl_handle, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt ($curl_handle, CURLOPT_POST, 1);
curl_setopt ($curl_handle, CURLOPT_POSTFIELDS,
'&method=CarrierPickupCancel&XMLInput=' . $xml);
$curl_result = curl_exec ($curl_handle) or die ("There has been a CURL_EXEC
error");
curl_close ($curl_handle);
echo $curl_result;
```

Example using SOAP submission (with PHP)

```
require_once('nusoap.php');
$soapclient = new soapclient('URL TO API');
$xml = XML (Formatted as above);
$params = array(
        'XMLInput' => $xml
                );
$soapclient->call('CarrierPickupCancel',$params);
echo $soapclient->response;
```

CarrierPickupCancel Error Codes

Error Code String	Error Message
Login Failed	Given in the root node, occurs if the account number or password is incorrect.
XMLInput Error	XML is not formatted correctly.
USPS Error Codes	No complete list available.

19. CHANGING A CARRIER PICKUP REQUEST API

CarrierPickupChange

Table 19-1: CarrierPickupChange XML Elements

Element	Type	Size or Values	Description
CarrierPickupChange	**Root**	**-**	**The root node for the XML request.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
PassPhrase	Text	64	Customer's Pass Phrase
Test	*Text*	*Y or N*	*Is this a test transaction?*
UseAddressOnFile	*Text*	*Y or N*	*Use your physical address we have on file (Default Y)*
FirstName	*Text*	*100*	*First name on the pickup (defaults to the first name on the account)*
LastName	*Text*	*100*	*Last name on the pickup (defaults to the last name on the account)*
CompanyName	*Text*	*100*	*Company name on the pickup (defaults to the company name on the account)*
SuiteOrApt	*Text*	*25*	*Suite or apartment number on the pickup (defaults to the one on file)*
Address	*Text*	*100*	*Address of the pickup (defaults to the one on file)*
City	*Text*	*100*	*City of the pickup (defaults to the one on file)*

Element	Type	Size or Values	Description
State	*Text*	*2*	*State abbreviation of the pickup (defaults to the one on file)*
ZIP5	*Numeric*	*5*	*Five digit ZIP Code of the pickup (defaults to the one on file)*
ZIP4	*Numeric*	*4*	*+4 ZIP Code for the pickup (defaults to the one on file)*
Phone	*Numeric*	*10*	*Phone number of the pickup (defaults to the one on file)*
Extension	*Numeric*	*5*	*Phone number extension for the pickup (defaults to the one on file)*
PackageLocation	Char	2	sd for Side Door kd for Knock on Door/Ring Bell mr for Mail Room of for Office rc for Reception im for In/At Mailbox ot for Other fd for Front Door bd for Back Door
SpecialInstructions	*Text*	*100*	*Required if "ot" is chosen for PackageLocation. Special Instructions for pickup.*
ConfirmationNumber	Text	100	The confirmation number of the pickup that you are changing
PickupList	Node	-	Container node for the Carrier Pickup List
PICNumber	*Numeric*	*30*	*Package PIC Number (Tracking Number)*
CustomsID	*Text*	*25*	*Package Customs ID*
PieceID	*Numeric*	*10*	*Package Piece ID*

The following listing shows an example of the CarrierPickupChange XML for the Carrier Pickup Change API.

Listing 19-1: CarrierPickupChange XML

```
<?xml version="1.0" encoding="utf-8"?>
<CarrierPickupChange>
        <AccountID></AccountID>
        <PassPhrase></PassPhrase>
        <Test></Test>
        <PackageLocation></PackageLocation>
        <ConfirmationNumber></ConfirmationNumber>
        <PickupList>
                <PICNumber></PICNumber>
                <PieceID></PieceID>
                <CustomsID></CustomsID>
        </PickupList>
</CarrierPickupChange>
```

CarrierPickupChangeResponse XML Syntax

The following table describes the XML tags for CarrierPickupChangeResponse.

Table 19-2: CarrierPickupChangeResponse XML Elements (XMLInput Parameter)

Element	Type	Size or Values	Description
CarrierPickupChangeResponse	**Root**	**-**	**The root node for the XML response.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
ErrorMsg	*Text*	-	*Error Message*
Test	*Text*	*Y or N*	*Is this a test transaction?*
Response	**Container Node**		**Container Node for the Response**
Error	*Container Node*		*Container Node For the Error (will only show IF there is an error)*
Number	*Numeric*		*USPS Error Code*
Description	*Text*	*250*	*USPS Error Code Description*
DayOfWeek	Text	Pickup Day of week (that is, “Monday”)	Pickup day of week
Date	Date	Xx/xx/xxxx	Date of pickup
CarrierRoute	Char	1	Pickup Route type (that is, “C” for City)
ConfirmationNumber	Text	100	Confirmation number of the pickup (Needed to change or cancel pickup)

The following listing shows an example of the CarrierPickupChangeResponse XML.

Listing 19-2: CarrierPickupChangeResponse XML

```
<?xml version="1.0" encoding="UTF-8"?>
<CarrierPickupChangeResponse>
        <AccountID></AccountID>
```

```
        <ErrorMsg></ErrorMsg>
        <Test></Test>
        <Response>
                <DayOfWeek></DayOfWeek>
                <Date></Date>
                <CarrierRoute></CarrierRoute>
                <ConfirmationNumber></ConfirmationNumber>
        </Response>
</CarrierPickupChangeResponse>
```

Using form submission: To use this API with a form submission you must declare both the method and the XML's parameter name in the submission.

Example:

```
<form name="CarrierPickupChange"
      method="POST"
      action="URL TO API&method=CarrierPickupChange">
      <input type="hidden" name="XMLInput" value="XML (Formatted as above)">
</form>
```

OR (through cURL in PHP):

```
$xml = XML (Formatted as above);
$curl_handle = curl_init ();

curl_setopt ($curl_handle, CURLOPT_URL, 'URL TO API');
curl_setopt ($curl_handle, CURLOPT_FOLLOWLOCATION, 1);
curl_setopt ($curl_handle, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($curl_handle, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt ($curl_handle, CURLOPT_POST, 1);
curl_setopt ($curl_handle, CURLOPT_POSTFIELDS,
'&method=CarrierPickupChange&XMLInput=' . $xml);
$curl_result = curl_exec ($curl_handle) or die ("There has been a CURL_EXEC
error");
curl_close ($curl_handle);
echo $curl_result;
```

Example using SOAP submission (with PHP)

```
require_once('nusoap.php');
$soapclient = new soapclient('URL TO API');

$xml = XML (Formatted as above);

$params = array(
        'XMLInput' => $xml
                );
$soapclient->call('CarrierPickupChange',$params);
echo $soapclient->response;
```

CarrierPickupChange Error Codes

Error Code String	Error Message
Login Failed	Given in the root node, occurs if the account number or password is incorrect.
XMLInput Error	XML is not formatted correctly.
You need at least one Express or Priority package	You need at least one Priority or Express mail shipment.
Invalid Package Location	You must use: sd for Side Door kd for Knock on Door/Ring Bell mr for Mail Room of for Office rc for Reception im for In/At Mailbox ot for Other fd for Front Door bd for Back Door
USPS Error Codes	**No complete list available.**

20. SCHEDULING A MULTIPLE LOCATION CARRIER PICKUP REQUEST API

MultiLocationCarrierPickup

Table 20-1: MultiLocationCarrierPickup XML Elements

Element	Type	Size or Values	Description
MultiLocationCarrierPickup	**Root**	**-**	**The root node for the XML request.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
PassPhrase	Text	64	Customer's Pass Phrase
Test	*Text*	*Y or N*	*Is this a test transaction?*

The following listing shows a syntax example of the CarrierPickupRequest XML for the Carrier Pickup Request API.

Listing 20-1: MultiLocationCarrierPickup XML

```
<?xml version="1.0" encoding="utf-8"?>
<MultiLocationCarrierPickup>
        <AccountID></AccountID>
        <PassPhrase></PassPhrase>
        <Test></Test>
</MultiLocationCarrierPickup>
```

MultiLocationCarrierPickupResponse XML Syntax

The following table describes the XML tags for CarrierPickupResponse.

Table 20-2: MultiLocationCarrierPickupResponse XML Elements (XMLInput Parameter)

Element	Type	Size or Values	Description
MultiLocationCarrierPickupResponse	**Root**	**-**	**The root node for the XML response.**
AccountID	Numeric	6	Account ID for the Endicia postage account.
ErrorMsg	*Text*	-	*Error Message*
Test	*Text*	*Y or N*	*Is this a test transaction?*
ScheduledPickups	**Container Node**		**Container Node for the Scheduled Pickups**

Element	Type	Size or Values	Description
GroupID	Text		Cost Center of the Pickup / Container Node of Pickup
CompanyName	Text		Company of pickup
Address	Text		Address of pickup
Name	Text		Name of person at pickup
City	Text		City of pickup
State	Char	2	State code of pickup
ZipCode	Numeric		ZIP Code of pickup
PhoneNumber	Text		Phone Number of pickup
EmailAddress	Text		Email address of pickup
ConfirmationNumber	Text	100	Confirmation number of pickup
Message	Text		Date of pickup
NotScheduled	**Container Node**		**Container node for the pickups not scheduled**
GroupID	Text		Cost center of the missed pickup / container node of missed pickup
CompanyName	Text		Company of missed pickup
Address	Text		Address of missed pickup
Name	Text		Name of person at missed pickup
City	Text		City of missed pickup
State	Char	2	State code of missed pickup
ZipCode	Numeric		ZIP Code of missed pickup
PhoneNumber	Text		Phone Number of missed pickup
EmailAddress	Text		Email address of missed pickup
ConfirmationNumber	Text	100	Confirmation number of missed pickup
Message	Text		Reason why it was not scheduled

The following listing shows an example of the MultiLocationCarrierPickupResponse XML.

Listing 20-2: MultiLocationCarrierPickupResponse XML

```
<?xml version="1.0" encoding="UTF-8"?>
<MultiLocationCarrierPickupResponse>
```

```
        <AccountID></AccountID>
        <ErrorMsg></ErrorMsg>
        <Test></Test>
        <ScheduledPickups>
                <GroupID>
                        <Name></Name>
                        <CompanyName></CompanyName>
                        <Address></Address>
                        <City></City>
                        <State></State>
                        <ZipCode></ZipCode>
                        <PhoneNumber></PhoneNumber>
                        <EmailAddress></EmailAddress>
                        <ConfirmationNumber></ConfirmationNumber>
                        <Message></Message>
                </GroupID>
        </ScheduledPickups>
        <NotScheduled>
                <GroupID>
                        <Name></Name>
                        <CompanyName></CompanyName>
                        <Address></Address>
                        <City></City>
                        <State></State>
                        <ZipCode></ZipCode>
                        <PhoneNumber></PhoneNumber>
                        <EmailAddress></EmailAddress>
                        <ConfirmationNumber></ConfirmationNumber>
                        <Message></Message>
                </GroupID>
        </NotScheduled>
</MultiLocationCarrierPickupResponse>
```

Using form submission: To use this API with a form submission you must declare both the method and the XML's parameter name in the submission.

Example:

```
<form name="MultiLocationCarrierPickupRequest"
      method="POST"
      action="URL TO API&method=MultiLocationCarrierPickup">
      <input type="hidden" name="XMLInput" value="XML (Formatted as above)">
</form>
```

OR (through cURL in PHP):

```
$xml = XML (Formatted as above);
$curl_handle = curl_init ();

curl_setopt ($curl_handle, CURLOPT_URL, 'URL TO API');
curl_setopt ($curl_handle, CURLOPT_FOLLOWLOCATION, 1);
curl_setopt ($curl_handle, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($curl_handle, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt ($curl_handle, CURLOPT_POST, 1);
curl_setopt ($curl_handle, CURLOPT_POSTFIELDS,
```

```
'&method=MultiLocationCarrierPickup&XMLInput=' . $xml);
$curl_result = curl_exec ($curl_handle) or die ("There has been a CURL_EXEC
error");
curl close ($curl handle);
echo $curl_result;
```

Example using SOAP submission (with PHP)

```
require_once('nusoap.php');

$soapclient = new soapclient('URL TO API');

$xml = XML (Formatted as above);

$params = array(
        'XMLInput' => $xml
                );

$soapclient->call('MultiLocationCarrierPickup',$params);


echo $soapclient->response;
```

CarrierPickupRequest Error Codes

Error Code String	Error Message
Login Failed	Given in the root node, occurs if the account number or password is incorrect.
XMLInput Error	XML is not formatted correctly.
Too Few Priority or Express	You need at least one Priority or Express mail shipment.
USPS Error Codes (Message Node)	**No complete list available.**

21. SIGNING UP FOR AN ACCOUNT API

UserSignupRequest

If you are developing software for use by multiple customers and thus will be maintaining multiple accounts, this chapter demonstrates an automated Web Service that will allow your customers to sign up for accounts more efficiently. You can collect all the needed information and just submit it through this API. In return, you will get an acknowledgement message back. The actual account number will still be returned through a separate email message and that will still need to be associated with the account in a separate step. This is necessary to maintain the security model of creating accounts.

> **Note:** Since January 2011, the USPS® has begun enforcing regulations requiring a "physical address" to be the actual location that your packages are prepared and shipped from, rather than P.O. Boxes™ and General Delivery addresses.

Important! The URL for this service is distinct and different than the Endicia Label Server Web Service URL. For more information, see The Endicia Label Server Web Service.

UserSignupRequest XML Syntax

The following table describes the XML tags for UserSignupRequest.

Important notes about all elements:

- Omitting any value will cause the Account Signup API to use the default value for that element.
- *Italic* items are optional.
- After signing up, the account number will be sent by email message to the customers email address. It will not be sent in the reply. This is for security reasons. In the event of the account number being lost, it can be found using the confirmation number sent in the reply.

Important notes about certain elements:

- **CompanyName**: The # symbol will be replaced with "No.".
- **PhoneNumber, PhoneNumberExtension, PhysicalZipCode, MailingZipCode, and FaxNumber:** Spaces, dashes and periods will be stripped.

Table 21-1: UserSignupRequest XML Elements (XMLInput Parameter)

Element	Type	Size or Values	Description
UserSignupRequest	**Root**	**-**	**The root node for the XML request.**
CompanyName	*Text*	*100*	*Name of the Customer's Company*
Test	*Text*	*Y or N*	*Is this a test request?*
FirstName	Text	25	Customer's First Name
MiddleInitial	*Text*	*25*	*Customer's Middle Initial*

Element	Type	Size or Values	Description
LastName	Text	50	Customer's Last Name
Title	*Text*	*50*	*Customer's Contact Title*
EmailAddress	Text	120	Customer's Email Address
EmailConfirm	Text	120	Customer's Email Address for error checking
PhoneNumber	Numeric	0-9 (xxx-xxx-xxxx)	Customer's Phone Number
PhoneNumberExtension	*Numeric*	*0-9*	*Customer's Phone Number Extention*
ICertify	Text	Y N	I certify that all information provided above is accurate and truthful. I also certify that I have read and understood the United States Postal Service Privacy Act Statement, PC Postage Privacy Principles,Endicia Terms and Conditions, Acknowledgement of Deposit, USPS Policy Regarding Shortpaid and Unpaid Postage.
FaxNumber	*Text*	*0-9*	*Customer's Fax Number*
PhysicalAddress	Text	200	Customer's Physical Address
PhysicalCity	Text	100	Customer's Physical City
PhysicalState	Text	2	Customer's Physical State (CA, MN, ND, SD, etc.)
PhysicalZipCode	Text	0-9 (xxxxx or xxxxx-xxxx)	Customer's Physical ZIP Code
MailingAddress	*Text*	*200*	*Customers Mailing Address (if different than physical)*

Element	Type	Size or Values	Description
MailingCity	*Text*	*100*	*Customers Mailing City (if different than physical)*
MailingState	*Text*	*2*	*Customers Mailing State (if different than physical)*
MailingZipCode	*Text*	*0-9 (xxxxx or xxxxx-xxxx)*	*Customers Mailing ZIP Code (if different than physical)*
WebPassword	Text	17	Customers Web Password (Needs to be 5-50 characters)
PassPhrase	Text	64	Customers Account Pass Phrase (Needs to be 5-50 characters)
ChallengeQuestion	Text	50	Question to be asked if the password is forgotten (Needs to be at least 5 characters)
ChallengeAnswer	Text	50	The answer to the Challenge Phrase (Needs to be at least 5 characters)
BillingType	*Text*	*TS*	*Type of billing (TS Default for LABELSERVER Product)*
PartnerId	Text	50	PartnerID (also called Requester ID) uniquely identifies the system making the request. This ID is assigned by Endicia.
ProductType	Text	LABELSERVER	Type of account you are signing up for
CreditCardNumber	Numerical	0-9	Credit Card Number
CreditCardAddress	Text	200	Address used with credit card
CreditCardCity	Text	100	City used for the credit card
CreditCardState	Text	2	State used with the credit card (CA, MN, ND, SD, etc.)
CreditCardZipCode	Text	0-9 (xxxxx or xxxxx-xxxx)	ZIP Code used with the credit card

Element	Type	Size or Values	Description
CreditCardType	Text	V (Visa) M (MasterCard) A (American Express) B (Carte Blanche) N (Discover/Novis) D (Diners Club)	Credit card value
CreditCardExpMonth	Numerical	01 02 03 04 … 12	Two digit representation of the credit card expiration month
CreditCardExpYear	Numerical	2005 2006 2007 …	Four digit representation of the credit card expiration year
PaymentType	Text	CC ACH	Credit Card or Checking Account (respectively)
CheckingAccountNumber	*Numerical*	*0-15*	*Checking account number (Required if Payment Type is ACH)*
CheckingAccountRoutingNumber	*Numerical*	*0-15*	*Routing Number of checking account (Required if payment type is ACH)*
OverrideEmailCheck	*Text*	*Y or N*	*Removes Duplicate Email Address Check.*

The following listing shows a syntax example of the UserSignupRequest XML for the Account Signup API.

Listing 21-1: UserSignupRequest XML (XMLInput Parameter)

```
<?xml version="1.0" encoding="utf-8"?>
<UserSignupRequest>
        <Test></Test>
        <CompanyName></CompanyName>
        <FirstName></FirstName>
        <MiddleInitial></MiddleInitial>
        <LastName></LastName>
        <Title></Title>
        <EmailAddress></EmailAddress>
        <EmailConfirm></EmailConfirm>
        <PhoneNumber></PhoneNumber>
        <PhoneNumberExtension></PhoneNumberExtension>
        <ICertify></ICertify>
        <FaxNumber></FaxNumber>
        <PhysicalAddress></PhysicalAddress>
        <PhysicalCity></PhysicalCity>
        <PhysicalState></PhysicalState>
        <PhysicalZipCode></PhysicalZipCode>
        <MailingAddress></MailingAddress>
        <MailingCity></MailingCity>
```

```
        <MailingState></MailingState>
        <MailingZipCode></MailingZipCode>
        <PassPhrase></PassPhrase>
        <PassPhrase></PassPhrase>
        <ChallengeQuestion></ChallengeQuestion>
        <ChallengeAnswer></ChallengeAnswer>
        <BillingType></BillingType>
        <PartnerId></PartnerId>
        <ProductType></ProductType>
        <CreditCardNumber></CreditCardNumber>
        <CreditCardAddress></CreditCardAddress>
        <CreditCardCity></CreditCardCity>
        <CreditCardState></CreditCardState>
        <CreditCardZipCode></CreditCardZipCode>
        <CreditCardType></CreditCardType>
        <CreditCardExpMonth></CreditCardExpMonth>
        <CreditCardExpYear></CreditCardExpYear>
        <PaymentType></PaymentType>
        <CheckingAccountNumber></CheckingAccountNumber>
        <CheckingAccountRoutingNumber></CheckingAccountRoutingNumber>
        <OverrideEmailCheck>N</OverrideEmailCheck>
</UserSignupRequest>
```

UserSignupResponse XML Syntax

The following table describes the XML tags for UserSignupResponse.

Table 21-2: UserSignupResponse XML Elements

Element	Type	Size or Values	Description
UserSignupResponse	**Root**	**-**	**The root node for the XML response.**
ErrorMsg	*Text*	*150*	*Error message*
ConfirmationNumber	Numerical	6	Confirmation Number of sign-up
Test	*Text*	*Y or N*	*Is this a test request?*
DateTime	Text	“mm/dd/yyyy – hh:mm:ss GMT”	Date and Time of Signup
ProductType	Text	LABELSERVER	Echoes your product

The following listing shows an example of the UserSignupResponse XML.

Listing 21-2: UserSignupResponse XML

```
<?xml version="1.0" encoding="UTF-8"?>
<UserSignupResponse>
        <ErrorMsg></ErrorMsg>
        <Test></Test>
        <ConfirmationNumber></ConfirmationNumber>
        <DateTime></DateTime>
        <ProductType></ProductType>
</UserSignupResponse>
```

Table 21-3: UserSignupResponse Error Codes

Error Code String	Error Message
XMLInput Error	XMLInput improperly formatted.
We have detected that you already have an account with this email address	Do you really want another account? Use <OverrideEmailCheck />.
You must include your first name	Error returned for missing node.
You must include your last name	Error returned for missing node.
You must include your email address	Error returned for missing node.
You must include the confirmation of your email address	Error returned for missing node.
You must include your phone number	Error returned for missing node.
You must state that you agree to our terms and conditions	Error returned for missing node.
You must include your physical address	Error returned for missing node.
You must include your physical city	Error returned for missing node.
You must include your physical state	Error returned for missing node.
You must include your physical zip code	Error returned for missing node.
You must include your Web password	Error returned for missing node.
You must include your pass phrase	Error returned for missing node.
You must include your challenge question	Error returned for missing node.
You must include your challenge answer	Error returned for missing node.
You must include the credit card number	Error returned for missing node.
You must include the credit card address	Error returned for missing node.
You must include the credit card city	Error returned for missing node.
You must include the credit card state	Error returned for missing node.
You must include the credit card zip code	Error returned for missing node.
You must include the credit card type	Error returned for missing node.
Physical Address Does Not Exist.	Error returned for invalid data.

Error Code String	Error Message
The address provided is not valid. Please note that USPS considers PO Box, Rural Route and General Delivery addresses as invalid physical addresses.	Error returned for invalid data.
Mailing Address Does Not Exist.	Error returned for invalid data.
You Must Agree to the Terms and Conditions	Error returned for invalid data.
You must enter your first name	Error returned for invalid data.
You must enter your physical address	Error returned for invalid data.
You must enter your physical city	Error returned for invalid data.
You must enter your physical state	Error returned for invalid data.
You must enter your physical zip code	Error returned for invalid data.
You have entered an invalid physical zip code, must be an integer	Error returned for invalid data.
You have entered an invalid physical zip code, must be 5 or 9 digits	Error returned for invalid data.
The Internet Password and the Pass Phrase cannot be the same.	Error returned for invalid data.
The Challenge Phrase and the Challenge Answer cannot be the same.	Error returned for invalid data.
Your Internet Password is too small. It must be between 5 and 17 characters.	Error returned for invalid data.
Your Pass Phrase is too small. It must be between 5 and 64 characters.	Error returned for invalid data.
Your Challenge Question is too small. It must be between 5 and 50 characters.	Error returned for invalid data.
Your Challenge Answer is too small. It must be between 5 and 50 characters.	Error returned for invalid data.
Credit card month invalid.	Error returned for invalid billing data.
You must include the credit card expiration month	Error returned for invalid billing data.
Credit card year invalid.	Error returned for invalid billing data.
You must include the credit card expiration year	Error returned for invalid billing data.
You must include the postage payment type	Error returned for invalid billing data.
You must include the product type	Error returned for invalid billing data.
Billing type not found	Error returned for invalid billing data.
Postage payment type not found, must be CC or ACH	Error returned for invalid billing data.
Please select a Credit Card Type.	Error returned for invalid billing data.
Please select a Credit Card Expiration Date.	Error returned for invalid billing data.
Please select a Credit Card Expiration Date.	Error returned for invalid billing data.
Invalid Expiration Year, the year cannot be less than the current year.	Error returned for invalid billing data.

Error Code String	Error Message
Invalid Expiration Month, the month cannot be less than the current month.	Error returned for invalid billing data.
Card number and card type do not match.	Error returned for invalid billing data.
Card number seems to be missing digits.	Error returned for invalid billing data.
Does not appear to be a supported card.	Error returned for invalid billing data.
Credit card number does not appear valid.	Error returned for invalid billing data.
Credit Card Number Can Only Contain Digits 0-9	Error returned for invalid billing data.
No credit card number entered.	Error returned for invalid billing data.
If paying by ACH you must include your checking account number	Error returned for invalid billing data.
Your checking account number seems to be invalid	Error returned for invalid billing data.
If paying by ACH you must include your routing number	Error returned for invalid billing data.
Your checking routing number seems to be invalid	Error returned for invalid billing data.

Using form submission: To use this API with a form submission you must declare both the method and the XML's parameter name in the submission.

Example:

```
<form name="UserSignupRequest"
      method="POST"
      action="URL TO API?method=UserSignup">
      <input type="hidden" name="XMLInput" value="XML (Formatted as above)">
</form>
```

OR (through cURL in PHP):

```
$xml = XML (Formatted as above);
$curl_handle = curl_init ();

curl_setopt ($curl_handle, CURLOPT_URL, 'URL TO API');
curl_setopt ($curl_handle, CURLOPT_FOLLOWLOCATION, 1);
curl_setopt ($curl_handle, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($curl_handle, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt ($curl_handle, CURLOPT_POST, 1);
curl_setopt ($curl_handle, CURLOPT_POSTFIELDS, '&method=UserSignup&XMLInput='
. $xml);
$curl_result = curl_exec ($curl_handle) or die ("There has been a CURL_EXEC
error");
curl_close ($curl_handle);
echo $curl_result;
```

Example using SOAP submission (with PHP)

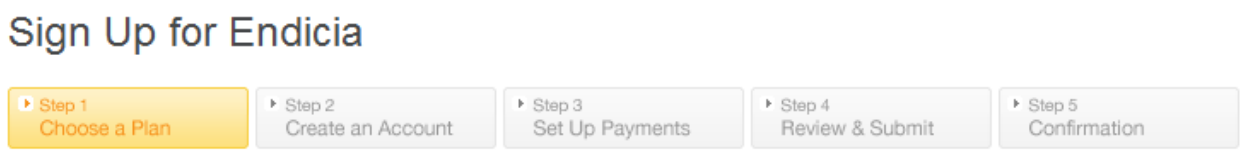
```
require_once('nusoap.php');

$soapclient = new soapclient('URL TO API');

$xml = XML (Formatted as above);

$params = array(
        'XMLInput' => $xml
                );

$soapclient->call('UserSignup',$params);

echo $soapclient->response;
```

Additional User Signup Options

Endicia.com Web site: This is appropriate for creating one account that the Integrator would use, but can be also be used to create many accounts for Partners. For example:

http://www.endicia.com/Labelserver/Login.cfm?partid=*xxxx*

where *xxxx* is the Partner ID assigned to you. Follow the screen prompts. This will take you through the signup process on our Web site:



The correct Partner ID *must* be included for the resulting account to be associated with you.

URL: Semi-automating the Web sign-up by passing parameters at the URL line: This allows the Partner to provide default values for some of the information on the Account sign-up pages. For example:

https://www.endicia.com/labelserver/login.cfm?PartID=*xxxx*

Required Components:

- PartID—Your Partner ID
- Optional Components:
- Company_Name—Your Company Name
- Contact_FName—Your First Name
- Contact_MName—Your Middle Initial
- Contact_LName—Your Last Name
- Contact_Title—Your Title
- Customer_Email—Your Email Address
- Confirm_Email—Your Email Address Again for Verification
- Physical_Address—Your Physical Address
- Physical_City—Your Physical City
- Physical_State—Your Physical State
- Physical_Zip_Code—Your Physical ZIP Code
- Mailing_address—Your Mailing Address
- Mailing_City—Your Mailing Address City
- Mailing_State—Your Mailing Address State
- Mailing_Zip_Code—Your Mailing Address ZIP Code

The URL with All Information:

https://www.endicia.com/labelserver/login.cfm?PartID=lngh&Company_Name=Length Co.&Contact_FName=Bob&Contact_MName=D&Contact_LName=Peterson&Contact_Title=CEO&Customer_Email=bob@Lengthco.com&Confirm_Email=bob@Lengthco.com&Physical_Address=123 Frank St.&Physical_City=New York&Physical_Zip_Code=12345

22. ADDITIONAL ENDICIA API'S

Dial-A-ZIP XML

As full-featured as Label Server is, there are limits to its ability to handle invalid addresses.

The Endicia Label Server Web Service returns a label if the destination address:
- is an exact match.
- has a valid city, state, and ZIP Code, but the street address is not a match.
- is a default match in the Dial-A-ZIP address database.

The Endicia Label Server Web Service returns an error if the destination address:
- is invalid.
- has an invalid city, state, or ZIP Code.
- is not found.
- has multiple matches, or is too ambiguous to resolve to a default address.

Dial-A-ZIP enables you to prevalidate and correct addresses before you send them to Label Server. Dial-A-ZIP uses the same logic as the U.S. Postal Service to correct the city, state and ZIP Code, as well as append the +4, and provides additional details.

For additional information on proper addressing, you can download USPS Publication 28, Postal Addressing Standards, at http://pe.usps.com/cpim/ftp/pubs/Pub28/pub28.pdf.

Endicia Printable Postage XML

The *Endicia Printable Postage* solution is available with your Endicia Label Server accounts. Endicia Printable Postage can print stamps without entering any address information. While Endicia Printable Postage is a client-based program, Endicia has an XML interface for Endicia Printable Postage rolls that can be printed through DYMO printers. This interface can be used in conjunction with Endicia Label Server as a solution when an address label is not required. More information is available upon request. For more information, see http://www.endicia.com/Features/InstaPostage/

Endicia XML

You may find that the Label Server API is more than you need for your shipping needs or that you need to integrate your shopping cart or online auction software into an XML solution that will work with DAZzle or Endicia for Mac. Available for PC or Mac, you can find more information at http://www.endicia.com/Developers/default.cfm.

Integrated Partners

Nearly 150 third-party software developers have integrated Endicia Label Server, Dial-A-ZIP, or Endicia XML into their own custom solutions. See the full list at:
http://www.endicia.com/Developers/IntegratedPartners/default_mvc.cfm.

23. PRINTING USING EPL2 AND ZPLII

For certain label layouts, the Label Server Web Service can generate label images in the Zebra native printer languages, EPL2 and ZPLII. The label image can contain both text and binary data. To ensure that the USPS can successfully scan the barcodes in the label, care must be taken to preserve the raw printer data when sending the label image to the Zebra printer.

> **Note:** To ascertain that the 2-D data matrix barcode is scanned properly, the format for the 2-D data matrix barcode will be migrated from the data stream type to the image type in version version 6.3.

For available EPL2 and ZPLII formats, see the LabelType, LabelSize, and MailpieceShape by MailClass section.

Some guidelines to follow when printing EPL2 and ZPLII labels to Zebra printers include:

- Verify that your Zebra printer has the appropriate data matrix firmware installed.

  The Label Server uses the data matrix symbology. If the Zebra printer does not contain the appropriate firmware, the square data matrix postage barcode will not print on the label, and the USPS will reject the mailpiece. The following is an EPL2 label with the data matrix barcode in the upper-right corner:

To upgrade the printer's firmware to support data matrix barcodes, see the Zebra support Web site:

http://www.zebra.com/id/zebra/na/en/index/drivers_downloads/firmware/firmware_for_printers.html

- Treat all data contained in the EPL2 or ZPLII label image as binary bytes

  The label image contains text and binary data values. All data must be treated as binary. Therefore, whether sending the label image data directly to the printer or saving it to intermediate storage for later retrieval, there should be no conversion performed to ANSI, ASCII or any other format. If that happens, the crucial barcode data bits will be changed causing the USPS scanners to read erroneous barcode data.

  The following example in Visual C# shows how to send raw data to a printer

```
public static bool SendBytesToPrinter(string szPrinterName, byte[] bytes)
{
```

```
    Int32 nLength;
    IntPtrpUnmanagedBytes;
    bool  bSuccess;

    pUnmanagedBytes = new IntPtr(0);
    nLength = Convert.ToInt32(bytes.Length);

    pUnmanagedBytes = Marshal.AllocCoTaskMem(nLength);

    Marshal.Copy(bytes, 0, pUnmanagedBytes, nLength);

    bSuccess = SendBytesToPrinter(szPrinterName, pUnmanagedBytes, nLength);

    Marshal.FreeCoTaskMem(pUnmanagedBytes);

    return(bSuccess);
  }

    public static bool SendBytesToPrinter(string szPrinterName,
                IntPtr pUnmanagedBytes, Int32 nLength)
  {
    DOCINFOA  di;
    Int32 dwError, dwWritten;
    IntPtrhPrinter;
    bool  bSuccess;

    bSuccess = false;

    dwError = 0;
    dwWritten = 0;

    di = new DOCINFOA();
    di.pDocName = "ELS Label";
    di.pDataType = "RAW";

    hPrinter = new IntPtr(0);
    if (OpenPrinter(szPrinterName.Normalize(), out hPrinter, IntPtr.Zero) == true)
    {
       if (StartDocPrinter(hPrinter, 1, di) == true)
       {
          if (StartPagePrinter(hPrinter))
          {
          bSuccess = WritePrinter(hPrinter, pUnmanagedBytes, nLength,
                        out dwWritten);
          EndPagePrinter(hPrinter);
          }
          EndDocPrinter(hPrinter);
       }
       ClosePrinter(hPrinter);
    }

    if (bSuccess == false)
    {
       dwError = Marshal.GetLastWin32Error();
    }

    return(bSuccess);
  }
```

- Correctly decode the sequence of data bytes in the label image.

The Zebra EPL2 and ZPLII languages have special sequences of bytes to represent special characters, such as a quote, back slash, and so on. These special sequences must be recognized and decoded properly to obtain a scannable label. Refer to the Zebra programming guides for further information.

24. CUSTOMS FORM OPTIONS

The Endicia Label Server offers full support for USPS-approved customs forms. Customs forms can be printed in full-size format or in a convenient 4x6 label format to streamline your workflow. Support for customs forms in Endicia Label Server includes the following capabilities:

- The addition of elements to populate fields on the *newest* USPS customs forms, including:
  - Restriction Type
  - Restriction Comments
  - Senders Customs Reference
  - License Number
  - Certificate Number
  - Invoice Number
  - Insured Number
  - HS Tariff Number
- ISO country codes for country of origin and for the ToCountryCode element
- Support for up to 999 customs items using continuation pages

To create a custom form, first choose the appropriate LabelType:

LabelType	Usage
International	For mailing to countries outside the USA
Domestic	For mailing to APO/FPO/DPO addresses or US Possessions and Territories

Choose LabelSubtype="Integrated", and choose the appropriate option for the IntegratedFormType element:

LabelType: International		
IntegratedFormType	**MailClass - MailpieceShape**	**Maximum CustomsItems[1]**
Form2976	FirstClassMailInternational; PriorityMailInternational (when used with FlatRateEnvelope, FlatRateLegalEnvelope, FlatRatePaddedEnvelope or SmallFlatRateBox)	5
Form2976A[2]	PriorityMailInternational (when used with Parcel, MediumFlatRateBox or LargeFlatRateBox); ExpressMailInternational (when used with FlatRateEnvelope, FlatRateLegalEnvelope or Parcel)	999, with continuation pages

[1] This limitation is due to form size constraints.
[2] 4×6 Form 2976A label accommodates a maximum of five customsItems.

For a list of valid country codes, see Country Code Table .

LabelType: Domestic		
IntegratedFormType	**MailClass—MailpieceShape**	**Maximum CustomsItems[1]**
Form2976	First, Priority, LibraryMail, MediaMail,	5

LabelType: Domestic		
IntegratedFormType	**MailClass—MailpieceShape**	**Maximum CustomsItems[1]**
	StandardPost	
Form2976A	First, Priority, LibraryMail, MediaMail, StandardPost	999, with continuation pages

[1] This limitation is due to form size constraints.

> **Note:** When using LabelType Domestic with LabelSubtype Integrated, MailClass Express, ParcelSelect, and StandardMail are not available.

What Qualifies as Domestic Mail

When shipping to domestic locations (US Territories and Possessions and Freely Associated States), use the ToState element with the State Code in the Domestic Column—these are not considered international, but may require a customs form.

U.S. Territories and Possessions (*Defined* as Domestic)	**Freely Associated States** (*Treated* as Domestic)
American Samoa (AS) • *Manua Island* • *Swain's Island* • *Tutuila Island* Guam (GU) Northern Mariana Islands, Commonwealth of the (MP) • *Rota Island* • *Saipan Island* • *Tinian Island* Puerto Rico, Commonwealth of (PR) * U.S. Virgin Islands (VI) * • *St. Croix Island* • *St. John Island* • *St. Thomas Island* Wake Atoll • *Wake Island*	Marshall Islands, Republic of the (MH) • *Ebeye Island* • *Kwajalein Island* • *Majuro Island* Micronesia, Federated States of (FM) • *Chuuk (Truk) Island* • *Kosrae Island* • *Pohnpei Island* • *Yap Island* Palau, Republic of (PW) • *Koror Island*
See http://pe.usps.com/text/dmm300/608.htm#1095140	See http://pe.usps.com/text/dmm300/608.htm#1127291

* No customs form required in most cases. See table at http://pe.usps.com/text/dmm300/608.htm#1237250.

Endicia Label Server applies appropriate pricing to military addresses when the ToCity element includes the correct Military Post Office designation:

- APO—Army Post Office
- FPO—Fleet Post Office
- DPO—Department of State Post Office

Military State Codes are defined as follows:

- AA—Armed Forces Americas (ZIP Codes 34xxx)

- AE—Armed Forces Europe, Canada, Middle East, Africa (ZIP Codes 09xxx)
- AP—Armed Forces Pacific (ZIP Codes 96xxx)

When a customs form is not required, you do not need to supply any customs information. Use LabelType="Default" as you would for any other U.S. address.

Populating the Customs Information

Use the CustomsInfo node to populate all the information required for your customs form.

Following is an example of an international Label Request using the new CustomsInfo node.

```
<LabelRequest Test="YES" LabelType="International"
     LabelSubtype="Integrated" ImageFormat="PNG" ImageResolution="300">
   <RequesterID>abcd</RequesterID>
   <AccountID>123456</AccountID>
   <PassPhrase>passphrase</PassPhrase>
   <MailClass>PriorityMailInternational</MailClass>
   <WeightOz>45</WeightOz>
   <MailpieceShape>PARCEL</MailpieceShape>
   <Services CertifiedMail="OFF" COD="OFF" DeliveryConfirmation="OFF"
     ElectronicReturnReceipt="OFF" InsuredMail="OFF" RegisteredMail="OFF"
     RestrictedDelivery="OFF" ReturnReceipt="OFF"
     SignatureConfirmation="OFF"/>
   <IntegratedFormType>FORM2976A</IntegratedFormType>
   <Description>International Label with Continuation Page</Description>
   <PartnerCustomerID>12345ABCD</PartnerCustomerID>
   <PartnerTransactionID>6789EFGH</PartnerTransactionID>
   <CustomsInfo>
      <ContentsType>Other</ContentsType>
      <ContentsExplanation>Merchandise</ContentsExplanation>
      <RestrictionType>NONE</RestrictionType>
      <RestrictionCommments />
      <SendersCustomsReference>Senders Ref</SendersCustomsReference>
      <ImportersCustomsReference>Importers Ref</ImportersCustomsReference>
      <LicenseNumber>License 123</LicenseNumber>
      <CertificateNumber>Cert No. 345</CertificateNumber>
      <InvoiceNumber>Invoice 678</InvoiceNumber>
      <NonDeliveryOption>RETURN</NonDeliveryOption>
      <EelPfc>NOEEI 30.37(a)</EelPfc>
      <CustomsItems>
         <CustomsItem>
            <Description>Item 1</Description>
            <Quantity>1</Quantity>
            <Weight>1</Weight>
            <Value>75</Value>
            <CountryOfOrigin>US</CountryOfOrigin>
         </CustomsItem>
         <CustomsItem>
            <Description>Item 2</Description>
            <Quantity>2</Quantity>
            <Weight>14</Weight>
            <Value>25.50</Value>
            <HSTariffNumber>123456</HSTariffNumber>
            <CountryOfOrigin>IE</CountryOfOrigin>
         </CustomsItem>
```

```
         <CustomsItem>
            <Description>Item 3</Description>
            <Quantity>3</Quantity>
            <Weight>25</Weight>
            <Value>27.50</Value>
            <CountryOfOrigin>DE</CountryOfOrigin>
         </CustomsItem>
      </CustomsItems>
   </CustomsInfo>
   <CustomsCertify>true</CustomsCertify>
   <CustomsSigner>Benjamin Franklin</CustomsSigner>
   <ResponseOptions PostagePrice="true" />
   <FromName>Benjamin Franklin</FromName>
   <FromCompany>United States Postal Service</FromCompany>
   <ReturnAddress1>475 L'Enfant PLZ SW</ReturnAddress1>
   <FromCity>Washington</FromCity>
   <FromState>DC</FromState>
   <FromPostalCode>20260</FromPostalCode>
   <FromPhone>2022681212</FromPhone>
   <ToCompany>Someone In Candada</ToCompany>
   <ToAddress1>Suite 400</ToAddress1>
   <ToAddress2>586 Argus Road</ToAddress2>
   <ToCity>Oakville</ToCity>
   <ToState>ON</ToState>
   <ToPostalCode>L6L 3J3</ToPostalCode>
   <ToCountryCode>CA</ToCountryCode>
   <ToPhone>8003877882</ToPhone>
</LabelRequest>
```

25. FREQUENTLY ASKED QUESTIONS

Answers in this section cover:

- Label Server Setup and Production
- Printing
- International Labels and Customs Forms
- Tracking

Label Server Setup and Production

Q. What does a customer need to do to move to the production environment?

A. A customer needs to send:

- A sample label that the customer has produced from the Test Server. Send the sample label by email to LabelServer@endicia.com.
- A signed agreement, including Exhibit B.

  Note: The agreement was sent to the customer by email, along with the technical specifications.

Q. How do I contact technical support?

A. Support is available by email message through LabelServer@endicia.com. If a phone call is required we can schedule one at that time.

Note: Include your XML request in any technical support requests when it is available.

Q. How do I reset my Pass Phrase in production?

A. You must reset the Pass Phrase through the API by using the ChangePassPhraseRequest method.

Note: Make sure your developer adds this feature because after you sign up for a new Endicia Label Server account, you will be required to reset the Pass Phrase for security purposes. Also note that starting in version 6.3, you cannot reuse up to four of your previously used Pass Phrases.

Q. If my account gets suspended due to bad login attempts, how can I reactivate my account?

A. To reactivate your account, send an email message to support at LabelServer@endicia.com, and then reset the Pass Phrase through the ChangePassPhraseRequest API. If you have implemented the API for Resetting Suspended Account Status, you can use it to reactivate your account.

Q. Do you offer customizable email notification options, like DAZzle does?

A. No, but you can send your customers an email notification based on your own business rules.

Q. When are dimensions required when submitting label requests?

A. Dimensions are required for Priority Mail items destined for Zones 5-8.

Purchasing Postage

Q. How do I purchase postage in production and what are my postage payment options?

A. You can only purchase postage through the API by using the RecreditRequest method. To review your postage payment options, see: http://www.endicia.com/USPSPaymentOptions.cfm.

Q. What are the requirements for submitting a refund for misprints or unused mailpieces?

A. All labels using the Delivery Confirmation, Signature Confirmation, or Express Mail tracking numbers must be submitted electronically. No physical submissions are allowed. To qualify, the item cannot be scanned by the USPS and must be requested within 10 days of printing.

Q. Is there a way to ensure that I have a postage account balance so that my account does not ever run out of funds?

A. The best way to ensure that you do not run out of funds is to write a statement to view the postage account balance that is included with every LabelRequestResponse, and if that postage account balance value is less than a specified dollar amount, subtract that postage account balance value from your postage account limit, and send a RecreditRequest with the amount being the result.

For example, if the postage account value is less than $100, subtract that value from $500 and submit the RecreditRequest for the difference.

Printing

Q. Do you offer a reprint option?

A. Reprinting is not available from the Endicia Label Server. You can either store the label image on your side or request a refund on the cancelled label and reprint.

Q. Having problems with printing the barcode on the label using Zebra printer?

A. If you are using the Zebra LP-2844 printer, you must upgrade the printer's firmware to support data matrix barcodes. See Printing using EPL2 and ZPLII.

Q. Can I add my company logo to the shipping label?

A. We do not offer it as part of the interface, partly because the overhead of sending the additional image back and forth may impact performance. However, we return your image of the label, on which you can superimpose your own logo.

Q. How do I convert a string containing the label image in Base64 format returned in the LabelRequestResponse XML into a bitmap object in .Net?

A. The following is a function in C# that converts a string containing the label image in Base64 format returned in the label request response into a bitmap object you can draw on the screen or print:

```
private Bitmap _GetBitmap(string base64LabelImage)
{
    byte[]       bytes = null;
    Bitmap       labelBitmap;
    MemoryStream memoryStream;
    bytes = Convert.FromBase64String(base64LabelImage);
    memoryStream = new MemoryStream(bytes);
    labelBitmap = (Bitmap) Image.FromStream(memoryStream);
    memoryStream.Close();
    memoryStream = null;
    return(labelBitmap);
}
```

Q. When I view a label on screen, sometimes the label is very large. When I print the label, the printing too big to fit on a label and sometimes spans more than one page.

A. If you are printing to a sheet-fed printer, you will need to set constraints on the size of the image. This can be due to variations in image resolution on your computer screen (usually 96 dpi), the image viewer you are using, or the printer's DPI resolution. When working with images, always set the image to be viewed in the same dimensions as requested. (4×6 inches, 4×5 inches, and so

on) and ensure the aspect ratio (proportions) are correct. Otherwise, USPS scanners will not be able to read the label.

A formula for determining the correct print size is:

Screen DPI * Desired Inches = Image Size in Pixels

Example: 96 screen DPI, label size 4×6:

96 * 6 = 576 pixels
96 * 4 = 384 pixels

International Labels and Customs Forms

Q. What are the options for shipping internationally?
A. First-Class Mail International, Priority Mail International, and Express Mail International are available for international shipping.

Q. What can you tell me about shipping to APO/FPO addresses that result in "Address not found"?
A. APO/FPO addresses are generally problematic. There is no official format that we are aware of. Missing or extra punctuation may cause addresses to be rejected by the address validation engine. For more information, see URL: http://www.penpalsforsoldiers.org/address.html

Q. Do APO/FPO shipments require customs forms?
A. According to the USPS, all mail going to an APO/FPO containing merchandise should include Customs Forms 2976-A. For more information, see: http://webapps.usps.com/customsforms/helppickaform.htm

For additional information on the how to generate customs forms in Label Server, see Customs Form Options.

Tracking

Q. Do you have a suggested site where we can send customers to view the status of their tracking number?
A. Refer your customers to either of the following:

- http://www.endicia.com/Status/?PIC=TRACKINGNUMBERHERE
- http://trkcnfrm1.smi.usps.com/PTSInternetWeb/InterLabelInquiry.do?origTrackNum=TRACKINGNUMBERHERE

Note: The second URL has much more current information.

26. LABELTYPE, LABELSIZE, AND MAILPIECESHAPE BY MAILCLASS

The following table shows the valid combinations that can be used with the available mail classes. Note that LabelType and LabelSize Attributes, and the MailpieceShape Element are all optional, and may be excluded from the request (or null) if you want the default result. For LabelType of Domestic or International, see Customs Form Options.

LabelType	MailpieceShape	LabelSize	Result	EPL/ZPL
First-Class Package Service and First-Class Mail Parcel				
Null or Default	Null or Parcel	Null or 4×6	4×6 Label	Y
		4×5	4×5 Label	Y
		4×4.5	4×4.5 Label	Y
		6×4	6×4 Label	N
		DocTab	Eltron 4×6.75 Label	Y
		4×8	4×8 Label	Y
CertifiedMail	Null, Letter, Flat, or Parcel	Null or 4×6	4×5.375 Label[1]	N
		7×4	7×4 Label	N
		8×3	8×3 Label	N
		Booklet	8.675x4.5 Label	N
		EnvelopeSize10	#10 Envelope	N
International[2]	Letter, Flat, or Parcel	N/A	Form2976 (CN 22)	Y[3]
Domestic	Flat or Parcel	N/A	Form2976 (CN 22)	Y[4]
First-Class Card, Letter and Flat				
DestinationConfirm	Card	6×4	6×4 Label	Y
	Letter or Flat	Null or 6×4	6×4 Label	Y
		7×3	7×3 Label	N
		Dymo30384	DYMO 30384 Label	N
		EnvelopeSize10	#10 Envelope	N
		Mailer7×5	7×5 Label	N
Priority Mail				
Null or Default	Null, Flat, Parcel, LargeParcel, IrregularParcel, and all RegionalRate and FlatRate [5] shapes.	Null or 4×6	4×6 Label	Y
		4×5	4×5 Label	Y
		4×4.5	4×4.5 Label	Y
		6×4	6×4 Label	N
		DocTab	Eltron 4×6.75 Label	Y
		4×8	4×8 Label	Y

[1] LabelSize is ignored.
[2] MailClass of FirstClassMailInternational
[3] No more than five customs items are supported when ImageFormat EPL2 or ZPLII are selected.
[4] Customs form not required for parcels less than 16 ounces.
[5] This includes the new FlatRateLegalEnvelope, FlatRatePaddedEnvelope, FlatRateGiftCardEnvelope, FlatRateWindowEnvelope, FlatRateCardboardEnvelope, and SmallFlatRateEnvelope.

CertifiedMail[6]	Flat or Parcel	Null or 4×6	4×5.375 Label[7]	N
		7×4	7×4 Label	N
		8×3	8×3 Label	N
		Booklet	8.675×4.5 Label	N
		EnvelopeSize10	#10 Envelope	N
International[8]	Null, Flat, Parcel, FlatRateEnvelope, FlatRateLegalEnvelope, FlatRatePaddedEnvelope, or Small, Medium, or Large FlatRateBox	Null	Form2976 (CN 22) 4×6 landscape	Y[9]
		4×6c	Form2976A (CP 72) landscape	Y
		Null	Form2976A (CP 72) 8.5×5.33	Y
Domestic	Null, Flat, Parcel, LargeParcel, IrregularParcel, and all RegionalRate and "FlatRate" [10] shapes.	N/A	Form2976 (CN 22), or Form2976A (CP 72)	Y[11]
Critical Mail				
Null or Default	Letter	Null	8.5×5.5 Image[12]	N
	Flat	Null	6×4 Image[13]	N
Express Mail				
Null or Default	Null, Letter, Flat, Parcel, MediumFlatRateBox or FlatRate and FlatRate Legal Envelope	Null or 4×6	4×6 Label	Y
		4×5	4×5 Label	Y
		4×4.5	4×4.5 Label	Y
		DocTab	Eltron 4×6.75 Label	Y
		4×8	4×8 Label	N
International[14]	Null, Flat, Parcel, FlatRateEnvelope or FlatRateLegalEnvelope	Null	Form2976 (CN 22) 4×6 landscape	Y[15]
		4×6c	Form2976A (CP 72) landscape	Y
		Null	Form2976A (CP 72) 8.5×5.33	Y

[6] Note that when using LabelType Default and choosing CertifiedMail from the Services node, MailpieceShape of Letter is not an available option for Priority Mail.
[7] LabelSize is ignored.
[8] MailClass of PriorityMailInternational
[9] Flat Rate Envelope or Small Flat Rate Box only.
[10] This includes the new FlatRateLegalEnvelope, FlatRatePaddedEnvelope, FlatRateGiftCardEnvelope, FlatRateWindowEnvelope, FlatRateCardboardEnvelope and SmallFlatRateEnvelope.
[11] Customs form not required for parcels under 16 ounces.
[12] LabelSize is ignored
[13] LabelSize is ignored
[14] MailClass of ExpressMailInternational
[15] Flat Rate Envelope or Small Flat Rate Box only.

Library Mail and Media Mail				
Null or Default	Null or Parcel	Null or 4×6	4×6 Label	Y
		4×5	4×5 Label	Y
		4×4.5	4×4.5 Label	Y
		6×4	6×4 Label	N
		DocTab	Eltron 4×6.75 Label	Y
		4×8	4×8 Label	Y
Domestic	Null or Parcel	N/A	Form2976 (CN 22), or Form2976A (CP 72)	Y
Standard Post and Parcel Select				
Null or Default	Null, Parcel or LargeParcel	Null or 4×6	4×6 Label	Y
		4×5	4×5 Label	Y
		4×4.5	4×4.5 Label	Y
		6×4	6×4 label	
		DocTab	Eltron 4×6.75 Label	Y
		4×8	4×8 Label	Y
Domestic[16]	Null or Parcel	N/A	Form2976 (CN 22), or Form2976A (CP 72)	N
Parcel Select LightWeight				
Null or Default	Null, Parcel, or IrregularParcel	Null or 4×6	4×6 Label	Y
		4×5	4×5 Label	Y
		4×4.5	4×4.5 Label	Y
		6×4	6×4 Label	N
		DocTab	Eltron 4×6.75 Label	Y
		4×8	4×8 Label	Y

[16] Not available for Parcel Select.

Appendix A. Change History

The following table shows the changes to the Endicia Label Server. The most recent changes appear first.

Date	Change
2013-01-27 Ver 7.0	▪ Version 7.0 ▪ The Parcel Post mail service is being renamed to Standard Post. ▪ The First-Class Mail International Parcel service is being renamed to First-Class Package International Service (FCPIS). First-Class Mail International Letters, Postcards, and Flats services remain unchanged. ▪ Electronic Delivery Confirmation is available for some international packages. ▪ The weight limit for International Flat Rate Envelopes will be changed from 20 pounds to 4 pounds. ▪ The previous limit of $100 guaranteed Express Mail International Insurance will be increased to $200. The price for the new, increased amount remains unchanged. ▪ Priority Mail International Flat Rate shipments to Mexico will no longer receive the same rate as previous mailings to Canada and Mexico. Shipments to Mexico will now be included with the "All Other Countries" price tier. All other Priority Mail International parcel services to Mexico will remain unchanged. ▪ Critical Mail will be available with a new signature option. This signature service will be added to the Critical Mail service at a more attractive rate than adding Signature Confirmation as an extra service. ▪ The label layouts of the Integrated Customs Forms PS-2976 and PS-2976A have been modified per USPS guidelines. ▪ Added <imageformat> values: “PNGMONOCHROME”, “JPEGMONOCHROME”, “GIFMONOCHROME”, and “BMPMONOCHROME”: The monochrome image format values can be used to specifically request monochrome labels that deliver up to an 85% reduction in file size for image file (BMP, JPEG, GIF, and PNG) formats. ▪ Added the ability to suppress fourth copy of the 4-part 2976A international customs form. ▪ Added two new values for <ContentsType> element: o HumanitarianDonations o DangerousGoods
2012-11-05 Ver 6.7.1	Version 6.7.1 ▪ Increased from 30 items to 999 items the maximum limit of customs form line items of the default 8.5”×5.5” international customs forms. ▪ Added an additional CostCenter element- "CostCenterAlphanumeric" in the "Requesting a Shipping Label" API to accommodate values that are alphanumeric. ▪ Fixed all issues related to USPS insurance for international packages. ▪ Fixed issues relating to the 4x6C International label layout, where the "Country of Origin" and "HS Tariff #" elements do not print if the customs items exceed 1 (one). ▪ Increased the size of the "PartnerCustomerID" element from 25 to 120 characters.
2012-09-05	Version 6.7

Date	Change
Ver 6.7	▪ Added ability to allow e-refunds for First Class Mail International Packages. ▪ Support for new 4”x 8” Doctab LabelSize. The 2” Doctab prints Tracking Number, Cost Center #, Reference ID and the three Rubber Stamps. ▪ Support for Registered Mail as a value added service. **Note:** Mailpiece must include an affixed USPS-issued Label 200. ▪ Support for new Open & Distribute containers with container-based pricing. o Supported Container Types: HalfTrayBox, FullTrayBox, EMMTrayBox, FlatTubTrayBox ▪ Support for Express Mail FlatRatePaddedEnvelope service. ▪ Support for Priority Mail® Large Flat Rate Board Game Box. New <MailPieceShape> value = ‘*LargeFlatRateBoardGameBox’.* ▪ Added support for 3 new reference fields in the GetLabelRequest() method: o <ReferenceID2> o <ReferenceID3> o <ReferenceID4> ▪ Support for calculating Live Animal Surcharge for shipping live animals such as day-old chicks. **Note:** The surcharge is automatically calculated based on weight and is only added if the destination zone is greater than 4, if the <LiveAnimalSurcharge> element is set to ‘Y’ ▪ Updated IMpb barcode from ¾” to 1” in height for all domestic labels. ▪ Fixed issues relating to the 4x6C Express Mail International label layout where the Country address line was being dropped off when the total number of address lines exceeds 3. ▪ Added new error codes: o 3027— ConsolidatorslabelsNotAllowed o 11803— DHLSortCodeLookupError
2012-05-29 Ver 6.6	<u>Version 6.6</u> ▪ Added new Web method—SubmitSharedPESUserContactInfo. ▪ Updated SCANRequestAPI to use <FromZipCode> as an optional filter for selecting tracking numbers to be included in a SCANRequest. ▪ Updated integrated 4x6 Customs form label layout: o To support up to 5 address lines in the "Ship To" address section. o The issue with HS Tariff number and country of origin not printing correctly for multiple lines in the EPL2 image format has now been fixed. o The issue with the insurance value printing incorrectly in the PS 2976-A long form has now been fixed. ▪ Integrated PS 2976 Customs form to U.S. Territories and Possessions are now allowed for all mail classes, irrespective of package weight ▪ The "Ship To" address block in the Express Mail label layout is now updated to support up to 5 address lines.

Date	Change
	▪ The individual item weight in the PS 2976 and PS 2976-A customs forms can be specified for a precision of up to 2 decimal places. ▪ The response for the LabelRequest has been modified to include the insurance fee in the <InsuredMail> element when Endicia Parcel Insurance is requested. The insurance fee is provided for visibility only. Customers will continue to be billed separately for the Endicia Parcel Insurance service. ▪ For Express Mail service, when additional services—such as COD, Hold For Pick Up, Adult Signature, and USPS insurance—are selected, the <SignatureWaiver> element will be ignored. ▪ Added new error codes: o 1067—SignatureRequired o 12144—InvalidPartnerCustomerId
2012-01-22 Ver 6.5.01	<u>Version 6.5.0.1</u> ▪ Formatted and re-arranged text.
2012-01-22 Ver 6.5	<u>Version 6.5</u> ▪ Added support for new USPS service – First Class Package Service. o Support for First-Class Mail Parcels will be retained as an option to mail "personal" packages weighing 13 oz or less at retail parcel prices. ▪ Added a new element <Pricing> to request for First Class Mail Parcel (Retail). Valid value: 'Retail' ▪ Added support for new mailpiece shape - Express Mail Medium Flat Rate Box (Dimensions: 11" x 8-1/2" x 5-1/2"). o To request an Express Mail Flat Rate Box, specify <MailClass> = 'Express' and <MailPieceShape> = 'MediumFlatRateBox'. ▪ Added support for new USPS Priority Regional Rate Box C. o To request a Priority Mail Regional Rate Box C, specify <MailClass> = 'Priority' and <MailPieceShape> = 'RegionalRateBoxC'. ▪ Added support for the new USPS Priority Mail Commercial Plus soft pack packaging alternative as an additional pricing option for cubic pricing. o Added a new <PackageTypeIndicator> element with available value = 'Softpack' for Priority Mail Commercial Plus cubic prices. o To request a cubic soft pack price, specify <MailClass> = 'Priority' and the new <PackageTypeIndicator> = 'Softpack'. ▪ Added support for a new Parcel Select mail category called Parcel Select Lightweight, that is a new mail presort (less than 1 pound) service. o To request this service, use the existing <MailClass> = 'ParcelSelect' with the appropriate <SortType> and <EntryFacility> values. ▪ Updated validation rules for Parcel Select Barcoded Non-Presort. The USPS has

Date	Change
	announced the following package validation changes to the Parcel Select Barcoded Non-Presort service: o Allowable weight limit increased from 35 lbs to 70 lbs o Machinability criteria changed from the current 34”x17”x17” to 27”x17”x17” and up to 35 lbs o Support for non-machinable and oversized rates o Free Delivery Confirmation service ▪ Changed the default value for the <SignatureWaiver> element to ‘TRUE’. ▪ Changed the current behavior of <DeliveryConfirmation> element. The Delivery Confirmation service will be added or excluded to a label response if the element is specified with ‘ON’ or ’OFF’ values in the label request. **Backward Compatibility Note:** The Delivery Confirmation service will be added or excluded to a label response if the element is specified with ‘ON’ or ’OFF’ values in the label request. However, in order to preserve backward compatibility, when the <Delivery Confirmation> element is NOT specified in the request, the Endicia Label Server will automatically add Delivery Confirmation service. For mail classes such as Library Mail, Media Mail, and Standard Post, mailers can choose to exclude Delivery Confirmation service by explicitly specifying the <DeliveryConfirmation> element set to ‘OFF’ in their Label Request API, thereby saving $0.19 per parcel for these mail classes. If no additional service including Delivery Confirmation has been requested, Endicia Label Server will still produce a label with the IMpb tracking barcode. No tracking will be available for this label through http://www.usps.com/ even though the banner text will show "USPS Tracking #" above the barcode. However, most the recent tracking information will be available at http://www.endicia.com or using the Requesting Package Status API. ▪ Added new error codes: o 1066 – SelectedServicesDoNotApply ▪ Removed <InsuredMail> value = “ON” from documentation. However, this value will still be available to support backward compatibility ▪ Miscellaneous Changes o Validation rules in the API's have been enhanced for attributes to only allow documented and valid values. For example, LabelType="" or DeliveryConfirmation="YES" would cause an error. o Positioning of certain attributes has been reorganized to support a more logical order. If customers are using an older version of the WSDL to perform deserialization, then the customers must upgrade to the most recent version of the WSDL.

Date	Change
	○ For LabelType = "International" or "DestinationConfirm", the only available service is <InsuredMail>, whereas previously other services were allowed.
2011-10-18 Ver 6.4	Version 6.4 ▪ Added support for 6×4 (landscape) layout for First-Class Flats, Letters, and Cards in high-speed EPL2/ZPLII format. ○ Supports up to 3 Rubber Stamps of 50 characters each. **Note:** For the sake of consistency, three Rubber Stamps have been added to the existing GIF, PNG and JPEG formats as well. ○ Added support for image formats: EPL2 and ZPLII. (PNG, JPEG, and GIF formats are already supported.) ▪ Added support for new USPS services: Adult Signature Required and Adult Signature Restricted Delivery. ▪ Implemented enforcement of USPS regulation to require a valid physical address rather than P.O.Box addresses for all new DYMO Endicia account signups. ○ If new customers sign up using a P.O.Box address, they will receive the following error: "The address provided is not valid. Please note that USPS considers PO Box, Rural Route, and General Delivery addresses as invalid physical addresses."
2011-08-02 Ver 6.3.1	Version 6.3.1 ▪ Added a new LabelSize attribute value "4×6c" for printing the new 4-part, 4"×6" international customs 2976-A forms for Priority Mail International and Express Mail International labels. **Note:** Null or any other value including "4×6" in the LabelSize attribute will return the default 8.5"×5.33" label.
2011-07-28 Ver 6.3	Version 6.3 ▪ Added support for 4×6, four-part, international customs 2976A form-set for the Priority Mail International and Express Mail International labels. ▪ Support for Planet Code will be deprecated as of this release. Planet Code is being replaced by IMb technology for DestinationConfirm services (First-Class Flats, Letters, and Cards). **Backward Compatibility Note:** The IMb tracking number returned in the response for the Label Request API will be up to 31 characters long. ▪ Going forward, the 2-D data matrix barcode for EPL2 and ZPLII label types will be printed in an image format instead of a data stream format. ▪ Package Status API: Added support for complete tracking of shipments: ○ If FullStatus element is specified in the request, the response will include a StatusBreakdown node that will provide complete status history for the shipment, beginning with the most recent status.

Date	Change
	Note: *Each Status element is named with an underscore "_" followed by an ascending number beginning with 1.* ▪ As part of an ongoing security compliance program, customer Pass Phrase history will now be tracked. As a result, new Pass Phrase cannot match any of the four previously used Pass Phrases. ▪ Improved support for international characters in the address section of labels in EPL2/ZPLII label formats. ▪ Removed error codes: o 11201—FailedRetrievingPlanetCode o 11202—PlanetCodeNumberServiceError o 11203—InvalidPlanetCodeNumberReceived o 11625—DialAZipStreetAddressNotMatch ▪ Added new error codes: o 11204—FailedRetrievingConfirmCode o 12669—PassPhraseReused ▪ Changed error code messages: o 2012—FailedRetrievingConfirmCode ▪ Added Coding Principles in Appendix C. ▪ Added a comprehensive LABELTYPE, LABELSIZE, and MAILPIECESHAPE BY MAILCLASS in section 26.
2011-06-02 Ver 6.2	<u>Version 6.2</u> ▪ ResetSuspendedAccount API: o Added new API to re-activate and reset the Pass Phrase of an account that has been suspended due to multiple retries. ▪ Added support for printing of *all* USPS-approved and eligible labels for customers with OMAS accounts. ▪ Transaction Listing API: o Added new element ReferenceID in the request and response ▪ Label Request API: o Added new elements ReferenceID and CostCenter in the response. ▪ SCAN Request API: o Added new element FormType; valid values: 5630 (default), 3152. o Added new element ReferenceID. o Updated scan instructions in the USPS Action section of the PS-5630 SCAN form USPS compliance purposes. ▪ PackageStatusRequest API: o Added new elements ReferenceID and CostCenter in the request and response. ▪ Updated the Customs form 2976 for APO/FPO/DPO addresses or US Posessions and Territories label layout, in compliance with USPS requirements:

Date	Change
	○ Customs Form2976 (short form) or Form2976A (long form) can be printed from the Label Request API by specifying the following options in the CustomsInfo node of the API: ○ LabelType—Domestic ○ LabelSubType—Integrated ○ IntegratedFormType—{Form 2976 or Form2976A} **Note:** For more information, see Customs form options. ▪ Added new error codes ○ 1064—ConflictingServices ○ 12004—VpoAdminCallFailed ○ 12508—IncorrectEmail ○ 12509—IncorrectChallengeAnswer ○ 100110—LabelNotAllowedForOfficialMail
2011-03-21 Ver 6.1	Version 6.1 ▪ Label Request API: ○ Added support for Priority Mail Open and Distribute and Express Mail Open and Distribute services. ○ Added new Services value for LabelRequest: OpenAndDistribute. ○ Added new elements to support Open and Distribute service: OpenAndDistributeFacilityType OpenAndDistributeFacilityName OpenAndDistributeTray OpenAndDistributeMailClassEnclosed OpenAndDistributeMailClassOther ○ Added Support for Return Receipt and Return Receipt Electronic as a combined service ▪ Label Request API and Calculate Postage and Fees API's: ○ Added new MailPieceShape of DVDFlatRateBox and LargeVideoFlatRateBox that will work with MailClass='PriorityMailInternational' ▪ Calculate Postage and Fees API's: ○ Added new element DateAdvance to support Date Advance feature in the Calculate Postage and Fees API's.
2011-01-02 Ver 6.0	Version 6.0 ▪ Label Request and Calculate Postage and Fees API's: ○ Added new MailClass of CriticalMail. ○ Added new MailpieceShape of FlatRateLegalEnvelope

Date	Change
	FlatRateGiftCardEnvelope, FlatRateWindowEnvelope, FlatRateCardboardEnvelope, SmallFlatRateEnvelope, RegionalRateBoxA and RegionalRateBoxB. ▪ Label Request: ○ Changed HoldForPickup from a ServiceLevel option to a Service node option, and removed Beta designation in Miscellaneous Notes and Features. ○ The TrackingNumber element is now obsolete. ○ Added HfpEmailAddress and HfpSMS elements to Label Request, and added HfpFacility elements to Label Response. Updated Miscellaneous Notes and Features accordingly. ▪ Label Server will return Error 2021 indicating that the user must certify that they agree to the Insurance Terms and Conditions when selecting Endicia in the InsuredMail attribute of the LabelRequest effective January 25, 2011. Customers must recertify every six months thereafter. ▪ Added introduction to Calculating Postage and Fees. ▪ Changed MailpieceShape tables for all Label Types to better display myriad of options in Chapter 19. ▪ Some labels will now include a four-digit Retail Distribution Code (RDC) on the right side of the label below the 2D barcode or to the left of it. This code helps the USPS separate mail more efficiently.
2010-10-15 Ver 5.3.3930	Version 5.3 ▪ Added support for Hold For Pickup service. ▪ Added new ServiceLevel value for LabelRequest: HoldForPickup. ▪ Added new elements to support Hold For Pickup service: HfpFacilityID HfpFacilityName HfpFacilityAddress1 HfpFacilityCity HfpFacilityState HfpFacilityPostalCode HfpFacilityZIP4 ▪ Added detailed explanation of International label options to Miscellaneous Notes and Features. ▪ Added compatibility table to Printing using EPL2 and ZPLII. ▪ Added updated support options to Contact the Endicia Label Server Team.
2010-07-06	Version 5.0

Date	Change
Ver 5.0.3826	▪ Added Continuation Page to international customs forms to allow up to 30 customs items. ▪ Added support for integrated labels with customs forms for international, APO/FPO, and other domestic destinations. These labels allow up to 30 customs items. ▪ **CustomsFormImageResolution** (LabelRequest XML) Deleted value: 150 (Obsolete) ▪ **CustomsInfo** (LabelRequest XML) New: To specify the customs declaration. ▪ **ImageResolution** (LabelRequest XML) Deleted value: 150 (Obsolete) ▪ **IntegratedFormType** (LabelRequest XML) New: To specify the form type to use for an integrated label. ▪ **LabelSubtype** (LabelRequest XML) New: To specify a subtype of a label type. ▪ **LabelType** (LabelRequest XML) Added new value: Domestic ▪ **ToCountryCode** (LabelRequest XML, PostageRateRequest XML, PostageRatesRequest XML) New: To specify the country code for an international address. ▪ Updated description of the following elements in the LabelRequest XML: **CustomsFormType, CustomsFormImageFormat, CustomsFormImageResolution** **OriginCountry, ContentsType, ContentsExplanation**, **NonDeliveryOption** **CustomsSigner** **ImageFormat** **ReplyPostage**, **ShowReturnAddress, TrackingNumber** **ReturnAddress3, ReturnAddress4, ToAddress3, ToAddress4** **ToCountry** ▪ Added new error codes: 1002 ElementNotAllowed 1050 NoCustomsItems 1055 ExceedsValueLimitForCustomsForm 1505 MailClassNotAllowedForLabel 11303 NoLabelNumberReceived 11304 InvalidLabelNumberReceived

Date	Change
	Backward Compatibility Note: Any element or value marked as obsolete in this version will no longer be supported after March 31, 2011. If your code uses these elements or values, the Endicia Label Server Web Service will continue to accept them until March 31, 2011.
2010-04-07 Ver 4.9.3741	Version 4.9 ▪ **CalculatePostageRates, CalculatePostageRatesXML** Added new Web methods to calculate "all" domestic or international postage rates.
2010-02-09 Ver 4.8.3691	Version 4.8 ▪ Added support for the updated USPS international and customs forms 2976-A. ▪ **EelPfc** (LabelRequest XML) New: To specify the EEL/PFC value for the customs forms 2976-A.
2009-12-24 Ver 4.7.3642	Version 4.7 ▪ Added support for USPS rates effective January 4, 2010. ▪ **MailpieceShape** (LabelRequest XML, PostageRateRequest XML) Added new value: FlatRatePaddedEnvelope ▪ **SortType** (LabelRequest XML, PostageRateRequest XML) Added new value: Nonpresorted. ▪ **EntryFacility** (LabelRequest XML, PostageRateRequest XML) Updated description. ▪ **MailpieceDimensions** (LabelRequest XML, PostageRateRequest XML) Updated description. ▪ **POZipCode** (LabelRequest XML) Updated description. ▪ **InsuredMail** (LabelRequest XML, PostageRateRequest XML) Increased the maximum insurable amount for Endicia Parcel Insurance to $10,000. ▪ **Country and Locality Listing** Added Kosovo. ▪ Added new error codes: 501 NoCubicPostageCalculator 502 CubicPricingNotOffered

Date	Change
	503 MailpieceCubicVolumeMustBePositive 504 MailpieceCubicVolumeExceedsMaximum 2020 AccountNotQualifiedForMailClass 2021 AccountNotQualifiedForFeature
2009-11-17 Ver 4.5.3603	Version 4.5 ▪ **MailClass** (LabelRequest XML, PostageRateRequest XML) Added new value: StandardMail ▪ **SortType** (LabelRequest XML, PostageRateRequest XML) Added new values: BMC, MixedBMC, SCF. ▪ **EntryFacility** (LabelRequest XML, PostageRateRequest XML) Updated description. ▪ **FromPostalCode** (PostageRateRequest XML) Updated description. ▪ **POZipCode** (LabelRequest XML) Updated description. ▪ **SundayHolidayDelivery** (LabelRequest XML, PostageRateRequest XML) New: To request Express Mail Sunday/Holiday Delivery Service. ▪ **ServiceLevel** (LabelRequest XML, PostageRateRequest XML) Deleted value: SundayHolidayDelivery **Backward Compatibility Note:** When ServiceLevel contains a value of SundayHolidayDelivery, the SundayHolidayDelivery element will be set to true.
2009-09-17 Ver 4.4.3544	Version 4.4 ▪ **International Mail labels (LabelRequest XML)** The tracking number for all international mail labels will be generated by Endicia. ▪ **FromName, FromCompany, ToName, ToCompany** Updated description. ▪ **SCAN Form API** The SCAN Form API now supports international labels with Endicia-generated tracking numbers.
2009-07-17 Ver 4.3.3485	Version 4.3 ▪ Added support for USPS NSA rates for qualified accounts. ▪ **FromPhone** (LabelRequest XML) Updated: Made required for international mail.

Date	Change
	▪ **MailpieceShape** (LabelRequest XML, PostageRateRequest XML) Added new value: MediumFlatRateBox Deleted value: FlatRateBox As of May 2009, USPS has renamed the regular Priority Mail Flat Rate box to Medium Flat Rate box. **Backward Compatibility Note:** When MailpieceShape contains FlatRateBox, it will be changed to MediumFlatRateBox. ▪ **ToPhone** (LabelRequest XML) Updated: Made optional for international mail. ▪ Added new error codes: 11002 PICNumberServiceError 11003 InvalidPICNumberReceived 11202 PlanetCodeNumberServiceError 11203 InvalidPlanetCodeNumberReceived 11451 FailedRetrievingAccountInformation 11452 AccountInformationServiceError 11453 InvalidValueInAccountInformation 11503 UnknownDialAZipReturnCode
2009-05-03 Ver 4.2.3404	Version 4.2 ▪ Added support for USPS domestic and international rates effective May 11, 2009. ▪ **LabelSize** (LabelRequest XML) To comply with USPS requirements, added a FIM to the indicia in EnvelopeSize10 and Postcard. ▪ **Services, ElectronicReturnReceipt** (LabelRequest XML) **BpodClientDUNSNumber** (LabelRequest XML) Updated description. ▪ **Stealth** (LabelRequest XML). Stealth cannot be used with LabelSize of EnvelopeSize10 and Postcard. ▪ **EmailOptions** (LabelRequest XML) Obsolete.
2009-03-20 Ver 4.1.3363	Version 4.1 ▪ **Express Mail labels (LabelRequest XML)** The tracking number for all domestic Express Mail labels will be generated by Endicia.

Date	Change
	▪ **Express Mail shipments** (SCAN Form Web Service API) SCAN form now accepts domestic Express Mail tracking numbers. ▪ **TrackingNumber** (LabelRequestResponse XML) Updated description.
2009-02-28 Ver 4.1.3344	Version 4.1 ▪ **CODAmount** (LabelRequest XML, PostageRateRequest XML) New: To specify COD amount. ▪ **EmailOptions** (LabelRequest XML) Obsolete: Beginning March 15th, 2009, Endicia Label Server will no longer support email notifications for Express Mail shipments. ▪ **EntryFacility** (LabelRequest XML, PostageRateRequest XML) New: To specify the entry facility. ▪ **FromPostalCode** (PostageRateRequest XML) Updated description. ▪ **InsuredValue** (LabelRequest XML, PostageRateRequest XML) Updated description. ▪ **MailClass** (LabelRequest XML, PostageRateRequest XML) Added new value: ParcelSelect ▪ **POZipCode** (LabelRequest XML) Updated description. ▪ **Services** (LabelRequest XML, PostageRateRequest XML) Added new attributes: COD, RestrictedDelivery ▪ **SortType** (LabelRequest XML, PostageRateRequest XML) Added new values: FiveDigit, Presorted ▪ **BarcodeFormat**, **FlatRate, NoHolidayDelivery**, **Oversize, PackageSize, ReturnToSender, TrackingType** (LabelRequest XML): Obsolete. ▪ **FlatRate, PackageSize** (PostageRateRequest XML)**:** Obsolete. ▪ **MailClass** (LabelRequest XML, PostageRateRequest XML) Obsolete values: INTLAIRPARCEL, INTLGEM, INTLGPM ▪ Added new error codes: 409 InvalidMailpieceValue 436 InvalidEntryFacility

Date	Change
	437 EntryFacilityDoesNotServiceDesination 11301 FailedRetrievingLabelNumber 11302 LabelNumberServiceError
2009-01-11 Ver 4.0.3296	Version 4.0 ▪ Added support for USPS domestic and international rates effective January 18, 2009. ▪ **GetPostageLabel, GetPostageLabelXML** Added support for USPS Commercial Plus pricing. These Web methods will use the Commercial Plus rate to calculate the postage based on the qualification of the account. **Note:** Accounts that qualify for Commercial Plus Pricing have been identified by the USPS and require no change in the label request. All other accounts will use Commercial Base Pricing by default for domestic Express Mail and Priority Mail service. ▪ **CalculatePostageRate, CalculatePostageRateXML** Added support for USPS Commercial Plus pricing. A new Pricing element has been added to the PostageRateRequest XML that can be used to request different levels of pricing. ▪ **IncludePostage** (LabelRequest XML) Allow IncludePostage value of FALSE for all mail classes. This capability allows you to print labels without postage. ▪ **MailpieceShape** (LabelRequest XML, PostageRateRequest XML) Added new value: SmallFlatRateBox ▪ **Pricing** (PostageRateRequest XML) New: To select pricing level. ▪ **PostagePrice** (LabelRequestResponse XML, PostageRateResponse XML) Updated: **Pricing** element within the **Postage** node to reflect new USPS pricing. ▪ **CostCenter** (GetTransactionsListing XML) A new element within the TrackingList node of the GetTransactionsListing API called <CostCenter> that can have a value of “All” or any numeric value to pull transactions with the specific CostCenter value. This is an optional value and the Cost Center will not be included in the response unless it has been added to the request.
2008-10-30	Version 3.55 ▪ **MailClass** (LabelRequest XML, PostageRateRequest XML)

Date	Change
Ver 3.55.3223	Added new value: FirstClassMailInternational ▪ **IncludePostage** (LabelRequest XML) Allow IncludePostage value of FALSE for international mail classes. ▪ **InsuredMail** (LabelRequest XML, PostageRateRequest XML) For international mail classes, the Label Server will return an error if Endicia Insurance is not available for the destination country. ▪ Added new error codes: 422 OnlinePostageNotAllowed 448 InsuranceNotAvailableToCountry ▪ E-refunds are not currently supported for FCMI through Label Server. We will provide support for physical refunds in the next update to the Endicia Label Server.
2008-09-12 Ver 3.54.3177	Version 3.54 ▪ **MailClass** (LabelRequest XML, PostageRateRequest XML) As of September 11, the USPS will require all Bound Printed Matter mailings to be paid by permit only. Therefore, BoundPrintedMatter will no longer be supported. ▪ **Stealth** (LabelRequest XML) Allow Stealth value of FALSE for international mail classes.
2008-07-22 Ver 3.53.3124	Version 3.53 ▪ **NonDeliveryOption** (LabelRequest XML) New: To specify nondelivery instructions. ▪ **ReturnToSender** (LabelRequest XML) Obsolete.
2008-07-06 Ver 3.52.3103	Version 3.52 ▪ **EmailOptions** (LabelRequest XML) New: To choose email notification option. ▪ **ImageFormat, CustomsFormImageFormat** (LabelRequest XML) Added new value: PDF
2008-06-06 Ver 3.51.3077	Version 3.51 ▪ **Postage** (PostageRateResponse XML) Updated description. ▪ **PostagePrice** (LabelRequestResponse XML, PostageRateResponse XML) New: To return the postage price.

Date	Change
	▪ **ResponseOptions** (LabelRequest XML, PostageRateRequest XML) New: To choose XML elements returned in the response.
2008-05-03	Version 3.5 Added support for USPS domestic and international rates effective May 12, 2008.
2008-04-30 Ver 3.4.3041	Version 3.4 ▪ **BarcodeFormat** (LabelRequest XML) Obsolete: The Endicia Label Server will choose the proper barcode format based on the label type. ▪ **InsuredMail** (LabelRequest XML, PostageRateRequest XML) Added new value: UspsOnline ▪ **ReplyPostage** (LabelRequest XML) Updated: Enabled support for additional label types. ▪ **Services** (LabelRequest XML, PostageRateRequest XML) Added new attribute: ReturnReceipt. ▪ Added new error codes: 449 ExtraServiceNotAvailable
2008-03-20 Ver 3.3.3001	Version 3.3 ▪ **NoHolidayDelivery** (LabelRequest XML) Obsolete. ▪ **NoWeekendDelivery** (LabelRequest XML) Updated: Endorsement will show as No Delivery Saturday. ▪ **ServiceLevel** (LabelRequest XML, PostageRateRequest XML) New: To set service level for Express Mail. ▪ **ShipDate, ShipTime** (LabelRequest XML, PostageRateRequest XML) New: To specify ship date and time for Express Mail Sunday/Holiday Delivery. ▪ Added new error codes: 2014 AccountPendingClose 2015 AccountIsClosed 11110 FailedRetrievingEMCommitments 11111 NotEligibleForSundayHolidayDelivery 11112 ServiceCommitmentsNotFound 11113 ServiceCommitmentNotOnSundayHoliday

Date	Change
	11114 CannotMeetServiceCommitmentCutOffTime 11115 NoSundayHolidayDeliveryToDestination
2008-03-01 Ver 3.2.2982	Version 3.2 Added support for the new USPS Priority Mail Large Flat Rate Box. ▪ **MailpieceShape** (LabelRequest XML, PostageRateRequest XML) Added new value: LargeFlatRateBox
2008-02-25	▪ SCANRequest, CarrierPickupRequest, CarrierPickupCancel, CarrierPickupChange, MultiLocationCarrierPickup, TransactionListingRequest, RefundRequest, StatusRequest ▪ Added PassPhrase node. ▪ **SCANRequest:** Added PDF as ImageFormat. ▪ **Get Transactions Listing API:** Corrected method name in documentation.
2008-02-08 Ver 3.1.2960	Version 3.1 ▪ As required by USPS regulations, the postage indicium in the label image returned in the LabelRequestResponse XML will contain the weight and zone information for all mailpieces which are classified as packages. ▪ **ImageResolution (LabelRequest XML)** Enabled support for 203 dpi for all image formats.

Date	Change
2008-01-08 Ver 3.1.2929	▪ **InsuredMail** (LabelRequest XML, PostageRateRequest XML) Enabled Endicia insurance for International mail. ▪ **InsuredValue** (LabelRequest XML, PostageRateRequest XML) New: To specify insured value. **Backward Compatibility Note:** When insurance is requested and InsuredValue is not supplied, it will be copied from the Value element. ▪ **LabelSize** (LabelRequest XML) Added new value: Mailer7×5 (for LabelType of DestinationConfirm) ▪ **Value** (LabelRequest XML, PostageRateRequest XML) Updated description.
2008-01-02	▪ **UserSignup** Changed length of CheckingAccountNumber and CheckingAccountRoutingNumber Nodes
2008-01-02	▪ **SCANRequest, CarrierPickupRequest, CarrierPickupCancel, CarrierPickupChange, MultiLocationCarrierPickup, TransactionListingRequest, RefundRequest, StatusRequest** Updated length of WebPassword node.
2007-12-13	▪ **UserSignup** Updated documentation to include remaining error messages Updated documentation to include notes about formatting of nodes. ▪ **SCANRequest, CarrierPickupRequest, CarrierPickupCancel, CarrierPickupChange, MultiLocationCarrierPickup** Added API
2007-11-09 Ver 3.1.2869	▪ **MailpieceShape** (LabelRequest XML, PostageRateRequest XML) USPS does not allow Delivery or Signature Confirmation service for First-Class Mail letter-size and flat-size mailpieces or Package Services flat-size mailpieces. To enforce this rule, the Endicia Label Server will now return one of the following error codes for an ineligible MailpieceShape and Confirmation Service combination: DeliveryConfirmationNotAvailable SignatureConfirmationNotAvailable
2007-10-18 Ver 3.1.2847	▪ **BarcodeFormat** (LabelRequest XML)

Date	Change
	Updated description. ▪ **ImageRotation (**LabelRequest XML) Enabled support for all rotation values. ▪ **MailClass** (LabelRequest XML, PostageRateRequest XML) Added new value: BoundPrintedMatter
2007-09-21	▪ **RefundRequest** Fixed documentation to show that the ErrMsg is returned regardless. Fixed documentation to show that it returns YES and NO, instead of True and False. ▪ **TransactionsListingRequest** Added From Address information to documentation.
2007-08-23 Ver 3.1.2791	▪ **LabelSize** (LabelRequest XML) Added new value: 6×4 (for LabelType of Default)
2007-08-02 Ver 3.1.2770	➢ Version 3.1 ▪ Added support for label output in ZPL II Programming Language: ▪ **ImageFormat** (LabelRequest XML). Added new value: ZPLII ▪ **LabelSize** (LabelRequest XML) Added new value for use with CertifiedMail label type: 8×3 ▪ **Results** (TransactionListingRequest) Added Results tag. ▪ **StartDate, StartTime, EndDate, and EndTime** (TransactionListingRequest) Made these tags optional. ▪ **TransactionListingRequest** Re-ordered output to be soonest transaction to oldest. Fixed title for status code definitions. ▪ **UserSignup** Added error messages. ▪ **UserSignup, RefundRequest, StatusRequest, GetTransactionsListing** Added <Test /> node to API Documentation.
2007-05-10 Ver 3.0.2686	▪ **ImageFormat** (LabelRequest XML). Enabled EPL2 for creating Priority Mail International, Flat Rate Envelope labels.

Date	Change
2007-05-03 Ver 3.0.2679	Version 3.0 Added support for the new 2007 USPS domestic and international rates. ▪ **CustomsCertify, CustomsSigner** (LabelRequest XML) New: To allow for electronic signing of international customs form.. ▪ **FlatRate**[1] (LabelRequest XML, PostageRateRequest XML) Obsolete. ▪ **Label** (LabelRequestResponse XML) New: To return the label as separate images. ▪ **LabelType** (LabelRequest XML) Added new value: International ▪ **MailClass** (LabelRequest XML, PostageRateRequest XML) Added new values: ExpressMailInternational [3] PriorityMailInternational [3] Deleted values: Postcard (*replaced by First*). IntlGEM [2] IntlGPM [2] IntlAirParcel [2] ▪ **MailpieceDimensions**[4] (LabelRequest XML, PostageRateRequest XML) New: To set the dimensions of the mailpiece. ▪ **MailpieceShape**[5] (LabelRequest XML, PostageRateRequest XML) Added new values: Card (*replaces Postcard*) FlatRateBox FlatRateEnvelope IrregularParcel LargeParcel OversizedParcel Deleted value: Postcard (*replaced by Card*). ▪ **Oversize**[1] (LabelRequest XML) Obsolete. ▪ **PackageSize**[1] (LabelRequest XML, PostageRateRequest XML) Obsolete.

Date	Change
	▪ **WeightOz** (LabelRequest XML, PostageRateRequest XML) Updated: Value allows for fractional ounces. **Backward compatibility notes:** 1. FlatRate, Oversize and PackageSize are obsolete and will no longer be supported after July 15, 2007. They have been superseded by MailpieceShape. If your code uses these elements, the Endicia Label Server Web Service will continue to map them to the appropriate MailpieceShape value. 2. IntlGEM, IntlGPM and IntlAirParcel will no longer be available starting May 14, 2007. If your code uses these mail classes, the Endicia Label Server Web Service will replace them with the new international mail classes: IntlGEM will be replaced with ExpressMailInternational; IntlGPM and IntlAirParcel will be replaced by PriorityMailInternational. 3. The label images for the new International mail classes have a different format. For example, Express Mail International has four parts compared to five in Global Express Mail. Priority Mail International Flat Rate Envelope has only one part compared to three for Global Priority Mail Flat Rate Envelope. 4. MailPieceDimensions is a new element that needs to be supplied for Priority Mail mailpieces going to Zones 5 – 8 and over 1 cubic foot to calculate the postage rate based on dimensional weight. Priority Mail Flat Rate Envelope and Priority Mail Flat Rate Box are not subject to this requirement. 5. MailpieceShape, if not supplied, will default to Parcel with the following exceptions: For First-Class DestinationConfirm and CertifiedMail labels, mailpiece shape will default to Letter if WeightOz is 3.5 oz or less; otherwise, it will default to Flat. For Priority Mail CertifiedMail labels, mailpiece shape will default to Flat.
2007-02-28 Ver 2.3.2615	▪ **LabelType** (LabelRequest XML) Destination Confirm labels require approval from USPS and Endicia. ▪ **PhoneNumberExtension** (UserSignupRequest XML) Updated: Fixed spelling, incorrectly spelled node still works. ▪ **StatusCode** (StatusRequestResponse XML)

Date	Change
	New: Added new node to the response. Updated: Fixed typographical errors in response definition. ▪ **GetTransactionsListing** (Endicia Label Server Web Service) Added new API to get all transactions matching the input.
2007-02-27 Ver 2.3.2614	▪ **RubberStamp1, RubberStamp2, RubberStamp3** (LabelRequest XML) Updated: Added Rubber Stamps to additional labels. ▪ **ToZIP4** (LabelRequest XML) Updated: Value is optional.
2007-02-21 Ver 2.3.2608	▪ **AutomationRate** (LabelRequest XML, PostageRateRequest XML) New: To use automation rates. ▪ **LabelSize** (LabelRequest XML) Added new value for use with DestinationConfirm label type: Dymo30384 ▪ **MailpieceShape** (LabelRequest XML, PostageRateRequest XML) New: To select shape of the mailpiece. ▪ **ReplyPostage (**LabelRequest XML) New: To print labels with reply postage. ▪ **ShowReturnAddress (**LabelRequest XML) New: To allow labels to print without the sender's address. ▪ **SortType** (LabelRequest XML, PostageRateRequest XML) New: To select sort level. ▪ **ValidateAddress (**LabelRequest XML) New: To bypass Dial-A-ZIP address validation.
2007-02-09 Ver 2.3.2596	▪ **ImageRotation (**LabelRequest XML) New: To allow rotation of label image. ▪ **LabelSize** (LabelRequest XML) Added new value for use with Default label type: DocTab
2007-01-30 Ver 2.3.2586	▪ **CalculatePostageRate, CalculatePostageRateXML** (Endicia Label Server Web Service) Added new Web methods to calculate postage rate. ▪ **ImageFormat** (LabelRequest XML). Removed support for BMP format. ▪ **LabelSize** (LabelRequest XML)

Date	Change
	Added new values for use with DestinationConfirm label type: EnvelopeSize10 Postcard ▪ **MailClass** (LabelRequest XML) Added new value: Postcard ▪ **RubberStamp1, RubberStamp2, RubberStamp3** (LabelRequest XML) New: To allow printing of user-supplied values. ▪ **Stealth** (LabelRequest XML) Updated: Defaults to TRUE. ▪ Sender's address is being validated; city, state and ZIP Code line is standardized.
2006-11-12 Ver 2.2.2507	▪ **GetAccountStatus, GetAccountStatusXML** (Endicia Label Server Web Service) Added new Web methods to get the status of an account.
2006-11-03 Ver 2.2.2498	▪ **FromCity, ToCity** (LabelRequest XML) City name can contain periods.
2006-10-29 Ver 2.2.2493	▪ **POZipCode (**LabelRequest XML) New: To specify mailing location. ▪ **FromName** (LabelRequest XML) Value is optional. ▪ **FromCompany** (LabelRequest XML) Value is required for Express Mail if FromName is omitted. ▪ **ToName** (LabelRequest XML) Value is optional. ▪ **ToCompany** (LabelRequest XML) Value is required for Express Mail if ToName is omitted. ▪ **RecreditAmount** (RecreditRequest XML) Added new predefined amounts: 7500, 10000, 20000.
2006-10-11 Ver 2.2.2482	▪ **ToPhone** (LabelRequest XML) Updated description.
2006-10-11 Ver 2.2.2475	▪ **RecreditAmount** (RecreditRequest XML) Field allows any amount of postage up to $99,999.99. ▪ Changed the Service Icon Text from "**1**" to "**FCM**" for First-Class Mail labels.

Date	Change
2006-09-20 Ver 2.2.2454	Version 2.2 ▪ Added support for customs forms: ▪ New in LabelRequest XML: **CustomsFormType** **CustomsFormImageFormat** **CustomsFormImageResolution** ▪ New in LabelRequestResponse XML: **CustomsForm** ▪ **LabelType** (LabelRequest XML) Added new value: CertifiedMail ▪ **LabelSize** (LabelRequest XML) Added new values for use with CertifiedMail label type: 4×6 7×4 Booklet EnvelopeSize10 ▪ **Base64LabelImageOptional** (LabelRequestResponse XML) Obsolete.
2006-08-27 Ver 2.1.2430	▪ Added support for Delivery Confirmation to APO/FPO addresses as well as selected Pacific Island ZIP Codes. ▪ Enabled Certified Mail and Electronic Return Receipt for Default label type.
2006-08-08 Ver 2.1.2411	▪ **LabelSize** (LabelRequest XML) Added new value: 6×4 (for LabelType of DestinationConfirm)
2006-07-24 Ver 2.1.2396	▪ **LabelType** (LabelRequest XML) Added new value: DestinationConfirm ▪ **LabelSize** (LabelRequest XML) Added new value for use with DestinationConfirm label type: 7×3
2006-07-19 Ver 2.1.2391	▪ **ContentsType** (LabelRequest XML) Added new value: ReturnedGoods ▪ **ContentsExplanation (**LabelRequest XML) New: To describe contents when ContentsType is set to OTHER. ▪ **BpodClientDunsNumber (**LabelRequest XML)

Date	Change
	New: To set DUNS number of partner.
2006-06-30 Ver 2.1.2372	▪ **FromCity, ToCity** (LabelRequest XML) City name can contain hyphens.
2006-06-28 Ver 2.1.2370	▪ **LabelType** (LabelRequest XML) Added new value: CertifiedMail
2006-05-23	▪ **MiddleInitial** (UserSignup XML) Changed Middle Initial to not required.
2006-05-17 Ver 2.1.2328	▪ **Services** (LabelRequest XML) Added new attribute: ElectronicReturnReceipt.
2006-05-16 Ver 2.1.2327	▪ **BarcodeFormat** (LabelRequest XML) Updated description. ▪ **ImageResolution (**LabelRequest XML) New: To allow selection of image resolution. ▪ **Services** (LabelRequest XML) Added new attribute: CertifiedMail. ▪ **TrackingType** (LabelRequest XML) Obsolete.
2006-04-06	▪ **PassPhrase** (LabelRequest XML, RecreditRequest XML, ChangePassPhraseRequest XML) Reduced size to 64 from 150. ▪ **NewPassPhrase** (ChangePassPhraseRequest XML) Reduced size to 64 from 150.
2006-03-22	Version 2.1 ▪ Added support for label output in EPL2 Programming Language: ▪ **ImageFormat** (LabelRequest XML). Added new value: EPL2
2006-02-07	Version 2.0

Date	Change
	▪ Added support for International Mail: ▪ Updated in LabelRequest XML: **MailClass** **ToPostalCode** **ToCountry** ▪ New in LabelRequest XML: **ReturnToSender** **OriginCountry** **ContentsType** **CustomsDescription1, CustomsQuantity1, CustomsWeight1, CustomsValue1, CustomsCountry1** **CustomsDescription2, CustomsQuantity2, CustomsWeight2, CustomsValue2, CustomsCountry2** **CustomsDescription3, CustomsQuantity3, CustomsWeight3, CustomsValue3, CustomsCountry3** **CustomsDescription4, CustomsQuantity4, CustomsWeight4, CustomsValue4, CustomsCountry4** **CustomsDescription5, CustomsQuantity5, CustomsWeight5, CustomsValue5, CustomsCountry5** ▪ New in LabelRequestResponse XML: **CustomsNumber** ▪ Added support for Planet Code labels (LabelRequest XML).
2006-01-11	▪ **IncludePostage** (LabelRequest XML) Added to allow Planet Code labels to print without postage.
2005-12-12	Version 1.2 ▪ Added support for current and new 2006 USPS domestic rates.
2005-08-15	▪ Bug fix (LabelRequest XML) Bug: Address verification would fail if the individual address elements of the Dial-A-ZIP XML were not URL encoded within the Endicia Label Server Web Service. Fix: URL encode the individual address elements of the Dial-A-ZIP XML.
2005-07-06	▪ **ToDeliveryPoint** (LabelRequest XML) Updated: Element is optional.

Date	Change
2005-07-05	▪ **LabelSize** (LabelRequest XML) Added new value: 4×4.5
2005-06-24	▪ LabelRequest XML The Endicia Label Server Web Service verifies and corrects, if necessary, the destination address. ▪ **MailClass** (LabelRequest XML) Added new value: LibraryMail ▪ **RecreditAmount** (RecreditRequest XML) Maximum amount of postage that can be added to the account has increased to $5,000.
2005-05-10	▪ **LabelSize** (LabelRequest XML) New: To select size of the label. ▪ **ToAddress1** (LabelRequest XML) Updated description.
2005-05-02	▪ Maximum insurable value for Endicia insurance has increased to $5,000.
2005-04-04	▪ **PackageSize** (LabelRequest XML) New: To set size of package to regular, large or oversize. ▪ **Oversize** (LabelRequest XML) Updated description.
2005-03-30	▪ Added support for the Stealth feature. ▪ Endicia Label Server Web Service calculates rates for “Local” zones properly. ▪ Returned label image in the LabelRequestResponse XML reflects the changes to address lines in the LabelRequest XML.

Date	Change
2005-03-24	▪ **ToName** (LabelRequest XML) Value is now optional except for Express Mail. ▪ **ToCompany** (LabelRequest XML) Value is required if ToAddress1 through ToAddress4 are omitted. ▪ **ToAddress1** (LabelRequest XML) Value is optional if ToCompany contains a value. ▪ **ToAddress5, ToAddress6, ReturnAddress5, ReturnAddress6** Obsolete. **Backward Compatibility Note:** If your code uses these obsolete tags, adjust accordingly. ▪ **ToName** **ToCompany** **ToAddress1, ToAddress2, ToAddress3, ToAddress4** **FromName** **FromCompany** **ReturnAddress1, ReturnAddress2, ReturnAddress3, ReturnAddress4** Reduced size to 47 characters from 150 characters. **Backward Compatibility Note:** If your code uses the extra characters for the name, company or delivery address lines, adjust accordingly.
2005-02-09	▪ **Machinable** (LabelRequest XML) New: To select machinable/non-machinable rate for Parcel Post. ▪ **Oversize** (LabelRequest XML) Updated to allow selection of oversize, non-machinable rate for Parcel Post.
2005-01-25	▪ **PIC** (LabelRequestResponse XML) Changed size of field to 22. The UCC/EAN Code128 barcode requires prepending of application identifier of "91" to the PIC number.
2005-01-04	▪ **BarcodeFormat** (LabelRequest XML) New: To support the UCC/EAN Code128 barcode format. ▪ **TrackingNumber** (LabelRequest XML) Changed size of PIC value to 22. The UCC/EAN Code128 barcode requires prepending of application identifier of "91" to the PIC number.

Appendix B. Error Codes

Error Code	Error Code String	Error Message
101	GenericError	**{Error Message}**
102	InternalEndiciaError	Internal Endicia Error: **{Error Message}**
307	InvalidOriginZipCode	Invalid origin ZIP Code.
308	InvalidDestinationZipCode	Invalid destination ZIP Code.
404	MailClassNotSupported	Mail class not supported (**{Mail Class}**).
405	InvalidMailClass	Invalid mail class (**{Mail Class}**).
406	MailpieceWeightMustBePositive	Mailpiece weight must be greater than zero.
407	MailpieceWeightExceedsMaximum	Weight (**{Weight}** ounces) exceeds maximum allowed for Mail Service (**{Weight}** ounces).
409	InvalidMailpieceValue	**{Description}** value is invalid.
410	MailpieceValueMustBePositive	**{Description}** value must be greater than $0.00.
411	MailpieceValueExceedsMaximum	Mailpiece value exceeds the **{Amount}** maximum allowed for **{Insurance Type}**.
412	MailpieceShapeNotSupported	Mailpiece shape not supported (**{Mailpiece Shape}**).
413	InvalidMailpieceShape	Invalid mailpiece shape (**{Mailpiece Shape}**).
414	ServiceLevelNotSupported	Service Level not supported (**{Service Level}**).
415	InvalidServiceLevel	Invalid Service Level (**{Service Level}**).
416	SortTypeNotSupported	Sort Type not supported (**{Sort Type}**).
417	InvalidSortType	Invalid Sort Type (**{Sort Type}**).
418	PricingNotOffered	**{Pricing}** pricing is not offered for **{Mail Service}**.
420	MailServiceNotAvailableToCountry	**{Mail Service}** is not available to **{Country Name}**.
421	InvalidCountry	**{Country}** is not a valid country.
422	OnlinePostageNotAllowed	Online postage is not allowed for **{Mail Service}** to **{Country Name}**.
423	LimitedMailServiceToCountry	**{Mail Service}** to **{Country Name}** is limited to **{allowed mail service(s)}**.
430	MailpieceDimensionsMustBePositive	Mailpiece length, width, and height must be greater than zero.
431	MailpieceDimensionsRequired	Mailpiece dimensions are required for **{Mail Class}**.
434	LengthPlusGirthExceedsMaximum	Mailpiece length plus girth of {**Dimension**}" exceeds maximum of {**Dimension**}" allowed by Mail Service.
436	InvalidEntryFacility	Invalid Entry Facility (**{Entry Facility}**).

Error Code	Error Code String	Error Message
437	EntryFacilityDoesNotServiceDesination	Entry Facility **{Entry Facility}** does not service destination ZIP code **{ZIP code}**.
440	DeliveryConfirmationNotAvailable	Delivery Confirmation is not available for **{Description}**.
441	SignatureConfirmationNotAvailable	Signature Confirmation is not available for **{Description}**.
448	InsuranceNotAvailableToCountry	**{Insurance Type}** is not available to **{Country Name}**.
449	ExtraServiceNotAvailable	**{Extra Service}** is not available for this mailpiece.
501	NoCubicPostageCalculator	Cubic postage calculator is not available for **{Mail Class} {Pricing}** pricing.
502	CubicPricingNotOffered	Cubic pricing is not offered for **{Mail Class} {Pricing}** pricing.
503	MailpieceCubicVolumeMustBePositive	Mailpiece cubic volume must be greater than zero.
504	MailpieceCubicVolumeExceedsMaximum	Cubic volume (**{Volume}** cu. ft.) exceeds maximum of **{Volume}** cu. ft. allowed for mail service.
1001	MissingOrInvalidXMLElement	Missing or invalid data element: **{Element Name}**
1002	ElementNotAllowed	Element not allowed: **{Element Name}**
1011	InvalidTrackingNumber	Service Type Code in Tracking Number must match Services requested for the package.
1020	InvalidRecreditAmount	RecreditAmount must be greater than zero and less than $100,000.
1050	NoCustomsItems	Customs Info does not contain any customs items.
1051	InvalidCustomsItemData	**{Element Name}** must be a positive numeric value **{Amount}** or less.
1052	TotalCustomsValueMustBePositive	Total customs value must be a positive numeric value 99999.99 or less.
1053	ContentsWeightExceedsPackageWeight	Total package weight must be equal to or greater than the sum of the item weights.
1054	ExceedsItemLimitForCustomsForm	**{Form Type}** is limited to only **{n}** items.
1055	ExceedsValueLimitForCustomsForm	Total customs value cannot exceed **{Amount}** for mailpiece using Customs Form **{Form Type}**.
1059	InvalidMailClassForOD	**{Mail Class}** is not a valid MailClass for open and distribute service.
1061	NoServicesAllowedForOD	No extra services are allowed with Open and Distribute service

Error Code	Error Code String	Error Message
1062	IncorrectWeightForOD	Enclosed mail cannot be placed directly in sacks and should be placed in tray(s), if package weight is less than 5lbs
1064	ConflictingServices	Conflicting services requested.
1066	SelectedServicesDoNotApply	Selected services **{Service 1, Service 2,…}** do not apply.
1067	SignatureRequired	Signature cannot be waived for Express Mail shipments when Adult Signature, COD, USPS insurance, or Hold For Pickup has been selected.
1101	AccountNotAuthorizedForLabel	Account #**{Account Number}** is not authorized for the requested label type..
1501	ExceedsLimitForAddressLines	Address lines exceed maximum allowed.
1505	MailClassNotAllowedForLabel	**{Mail Class}** is not allowed for label.
2010	InvalidAccountStatus	Account #**{Account Number}** has an invalid status. Please contact the Endicia Label Server Support Team.
2011	AccountNeedsPassPhraseReset	Account #**{Account Number}** requires a Pass Phrase reset.
2012	AccountSuspendedBadLogins	Account #**{Account Number}** has been suspended due to multiple login attempts with an incorrect Pass Phrase. Please contact Endicia customer support at (650) 321-2640 x130 or LabelServer@endicia.com.
2013	AccountSuspendedForCause	Account #**{Account Number}** has been suspended. Please contact Endicia customer support at (650) 321-2640 x130 or LabelServer@endicia.com.
2014	AccountPendingClose	Account #**{Account Number}** is scheduled to be closed. Please contact Endicia customer support at (650) 321-2640 x130 or LabelServer@endicia.com.
2015	AccountIsClosed	Account #**{Account Number}** is closed. Please contact Endicia customer support at (650) 321-2640 x130 or LabelServer@endicia.com.
2020	AccountNotQualifiedForMailClass	Account #**{Account Number}** is not qualified for **{Mail Class}**.
2021	AccountNotQualifiedForFeature	Account #**{Account Number}** is not qualified for **{Feature Description}**.
3005	NameOrCompanyRequired	**{Sender's | Recipient's}** name or company required.
3010	InsuredValueExceedsEndiciaMaximum	Insured value exceeds the $10,000 maximum allowed by Endicia.
3011	InsuredValueMustBePositive	Insured value must be greater than $0.00.

Error Code	Error Code String	Error Message
3020	FailedValidatingShipFromAddress	Unable to validate Ship From address.
3021	FailedValidatingShipToAddress	Unable to validate Ship To address.
3023	ExpressMailNotAvailableToDestination	Express Mail Service is not available to the destination address.
3024	ServiceNotAvailableToDestination	One or more services are not available to the destination address.
3025	CustomsFormNotAllowedForMailpiece	Customs Form is not allowed for mailpiece.
3027	ConsolidatorslabelsNotAllowed	Requested Consolidator Label is invalid.
4001	InvalidFieldInDialAZipResponse	The **{Field Name}** field in Dial-A-ZIP response is invalid.
11001	FailedRetrievingPICNumber	Unable to retrieve PIC number: **{Description}**
11002	PICNumberServiceError	Error returned from PIC Number Service.
11003	InvalidPICNumberReceived	Invalid PIC number received: **{Number}**
11101	FailedRetrievingExpressMailNumber	Unable to retrieve Express Mail number: **{Description}**
11102	ExpressMailAPIError	Error returned from Express Mail API: **{Description}**
11110	FailedRetrievingEMCommitments	Unable to retrieve Express Mail Service Commitments: **{Description}**
11111	NotEligibleForSundayHolidayDelivery	Mailpiece not eligible for Express Mail Sunday/Holiday Delivery Service.
11112	ServiceCommitmentsNotFound	There were no Express Mail Service Commitments for the mailpiece.
11113	ServiceCommitmentNotOnSundayHoliday	Express Mail Sunday/Holiday Delivery Service not allowed: Service Commitment does not fall on a Sunday or a Holiday.
11114	CannotMeetServiceCommitmentCutOffTime	Ship time is past the latest cut-off time to meet Express Mail Service Commitment.
11115	NoSundayHolidayDeliveryToDestination	Express Mail Sunday/Holiday Delivery Service is not available to ZIP Code **{Zip Code}**.
11204	FailedRetrievingConfirmCode	Unable to retrieve IMb confirmation code: **{Description}**
11301	FailedRetrievingLabelNumber	Unable to retrieve **{Mail Class/Extra Service}** number: **{Description}**
11302	LabelNumberServiceError	Error returned from Label Number Service: **{Description}**
11303	NoLabelNumberReceived	Label Number Service did not return a label number.
11304	InvalidLabelNumberReceived	Invalid label number received from Label Number Service: **{Label Number}**
11401	FailedRetrievingBarcodeNumber	Unable to retrieve **{Description}** barcode number.

Error Code	Error Code String	Error Message
11402	InternationalLabelAPIError	Error returned from International Label API: **{Description}**
11403	CustomsFormsAPIError	Error returned from Customs Forms API: **{Description}**
11451	FailedRetrievingAccountInformation	Unable to retrieve information for account #**{Account Number}**: **{Description}**
11452	AccountInformationServiceError	Error returned from Account Information Service for account #**{Account Number}**: **{Description}**
11453	InvalidValueinAccountInformation	Invalid **{Value Name}** value received for account #**{Account Number}**.
11501	FailedValidatingAddress	Unable to validate address: **{Description}**
11502	NonDeliverableAddress	The address is not a deliverable location according to the US Postal Service.
11503	UnknownDialAZipReturnCode	Dial-A-ZIP Server ({**Server Name**}) has returned an unknown return code of {**value**}.
11611	DialAZipInvalidZipCode	Dial-A-ZIP error: Invalid ZIP Code
11612	DialAZipInvalidStateCode	Dial-A-ZIP error: Invalid state code.
11613	DialAZipInvalidCity	Dial-A-ZIP error: Invalid city.
11621	DialAZipAddressNotFound	Dial-A-ZIP error: Address not found.
11622	DialAZipMultipleMatchesTooAmbiguious	Dial-A-ZIP error: Address is too ambiguous.
11803	DHLSortCodeLookupError	Failed to retrieve DHL Sort code: {DHL sortcode API error message}.
12001	PurchaseTooSmall	Buy transaction must be greater than $10.00.
12002	IndiciumValueZero	Indicium value is zero.
12003	VpoTransactionFailed	VPO transaction failed (**{Status Codes}**): **{Description}**
12004	VpoAdminCallFailed	VPO Admin Transaction failed (**{Status Codes}**): **{Description}**
12100	UnknownPostageTransactionError	The Postage Transaction has returned an unknown error.
12101	AccountNotFound	The Certified Intermediary's account number is invalid.
12103	InactiveAccount	The Certified Intermediary's account is not active.
12104	InsufficientFunds1	The Certified Intermediary's account does not have enough postage balance to support this transaction.
12115	IndiciumMaxExceeded	The indicium postage requested exceeds the account's limit
12116	InvalidCreditCard	Invalid credit card on account.

Error Code	Error Code String	Error Message
12117	InvalidCCExpiration	The account credit card's expiration date has passed
12118	InvalidPurchaseAmountLow	The purchase amount is too low.
12119	InvalidPurchaseAmountHigh	The purchase amount exceeds the account's purchase limit.
12120	CreditCardFailed	The credit card transaction failed.
12121	PurchaseUpdateFailed	The update to the account with the credit card information failed.
12124	DatabaseFailure	General database error
12125	InvalidRequest	The XML Request type is not recognized
12128	InvalidCCDeclined	The credit card purchase was declined by the customer's bank.
12129	InvalidCCReferred	The credit card purchase was "referred" by the customer's bank. Contact the bank.
12130	Unavailable	The Postage Server is unavailable.
12131	InvalidCCAddress	The address on the credit card is invalid.
12132	CCUnavailable	The credit card processing is unavailable.
12133	Locked	The account is locked (possibly in use).
12134	AccountHoldFailed	The request to hold the account for a transaction failed.
12135	InternetAccessFailure	Internet access/communication failure.
12136	AccountIsBusy	Account is busy.
12144	InvalidPartnerCustomerId	PartnerCustomerID is missing or Invalid.
12201	XmlParseError	The XML request cannot be parsed for all required elements. The description provides details on the problem item.
12500	KeyMacCheckFailed	The key on the account has been tampered with and no transactions are allowed.
12502	NegativeDollarIndiciumRequest	Indicium requests must be positive.
12503	InsufficientFunds2	There is not enough money in the account to produce the indicium.
12505	AccountMacFailed	The account information has been tampered with and no transactions are allowed.
12507	IncorrectPassPhrase	The Certified Intermediary's Pass Phrase is incorrect.
12508	IncorrectEmail	The email provided is incorrect.
12509	IncorrectChallengeAnswer	The challenge answer provided is incorrect.

Error Code	Error Code String	Error Message
12514	ReplayDetected	The system has detected a duplicate request for postage.
12515	AccountMacCheckFailed	The account information has been tampered with and no transactions are allowed.
12525	AccountSuspended	The account has been suspended and no transactions are allowed.
12669	PassPhraseReused	The new Pass Phrase cannot match up to the four previously used Pass Phrases.
13001	ConfirmReceiptFailed	Internal Error: Confirm Receipt transaction failed for label number {0}.
100002	InternalLabelServiceError	Internal Endicia Label Server Web Service Error: **{Error Message}**
100003	InvalidSoapRequestFormat	The format of the SOAP request is invalid.
100105	FailedCreatingPostageLabel	Failed to create postage label.
100106	FailedCreatingCustomsForm	Failed to create customs form.
100110	LabelNotAllowedForOfficialMail	Label is not allowed for official mail.
100201	AccountNotQualifiedForPricing	Account #**{Account Number}** is not qualified for **{Mail Class}** **{Pricing}** pricing.

Appendix C. More Source Examples

This appendix contains various examples of source material to demonstrate how to perform certain actions, such as requesting a shipping label, including:

- Coldfusion Examples
- C# Examples
- Perl Examples
- PHP Examples
- Ruby Examples
- VisualBasic.NET Examples

Coding Principles

When developing an end user or customer solution, remember that the application will be making financial transactions. The USPShas many rules regarding the purchase of postage (RecreditRequest), the printing of postage (LabelRequest), and obtaining refunds (RefundRequest). Therefore, it would be prudent to design the application in such a way that basic protections are in place, such as:

- Protection from transmitting multiple Label Requests from a single label. Failing to prevent an end user or customer from clicking multiple times, or otherwise transmitting the same transaction multiple times, could cause the account to be charged multiple times for a single transaction. Refunds may be difficult to obtain without physical evidence.

- Storage of the Base64 or decoded image until the label has physically been printed and used on a shipment. It may be necessary to reprint a misprinted label, or a label that failed to print at all. Also, some refund requests may require physical evidence, so the label may need to be printed.

- A RecreditRequest (postage purchase) cannot result in the balance exceeding the maximum allowable balance set for the account (usually $500). If you build a trigger into your code that initiates a postage purchase, base it on a preset balance amount, and check each LabelRequestResponse (the PostageBalance element). When the balance is below your preset threshold, initiate the RecreditRequest so that it will only purchase the amount necessary to bring the balance to the maximum allowable balance.

- Retrieving values from XML elements should be based on the element or attribute name, not on a count or other means that does not validate that the correct value is returned. On occasion, elements and attributes may be added to one or more of our Web methods which could change the position of an element or attribute relative to the other elements in the response.

Caution Statements

Note the following:

- No support or service of these samples is implied or acknowledged and use of it is completely at the user's discretion and responsibility.

- This section contains sample application programs in source language that illustrate programming techniques on various operating platforms. These examples have not been thoroughly tested under all conditions. Endicia, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

- The source code provided here are for sample use and are provided "as-is"; Endicia disclaims all express and implied warranties, including, the implied warranties of merchantability or fitness for a particular use. Endicia shall not be liable for any damages, including, direct, indirect, incidental, special or consequential damages for loss of profits, revenue, data or data use, incurred by you or any third party in connection with the use of these materials.

- This is a sample application and is provide "as is" and without warranty. The technical approach used in this sample implementation is one approach to achieve integration between the Endicia Label Server Web Service and a system. The techniques presented may not be appropriate for your system, for all scenarios, and within all development environments. For more information, please make sure you consult the associated documentation of your development environment.

Coldfusion Examples

This section shows one or more Coldfusion source examples:

- Request a First-Class Shipping Label: LabelServiceInvoke.cfm

Request a First-Class Shipping Label: LabelServiceInvoke.cfm

The following source example:

- Requests a First-Class shipping label, using the Request a Shipping Label API.
- Uses the ColdFusion development environment.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<cfsavecontent variable="soap_packet">
<cfoutput>
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetPostageLabel xmlns="www.envmgr.com/LabelService">
      <LabelRequest ImageFormat="JPEG" Test="YES">
      <RequesterID>TESTREQ</RequesterID>
      <AccountID>123456</AccountID>
      <PassPhrase>PASSWORD</PassPhrase>
      <MailClass>FIRST</MailClass>
      <WeightOz>1</WeightOz>
      <PartnerCustomerID>CUSTID</PartnerCustomerID>
      <PartnerTransactionID>1000</PartnerTransactionID>
      <ToName>Benjamin Franklin</ToName>
      <ToCompany>United States Postal Service</ToCompany
      <ToAddress1>475 L'Enfant Plaza SW</ToAddress1>
      <ToCity>Washington</ToCity>
      <ToState>DC</ToState>
      <ToPostalCode>20260</ToPostalCode>
      <FromName>Technical Support</FromName>
      <FromCompany>DYMO Endicia</FromCompany>
      <ReturnAddress1>385 Sherman Ave</ReturnAddress1>
      <FromCity>Palo Alto</FromCity>
      <FromState>CA</FromState>
      <FromPostalCode>94306</FromPostalCode>
      </LabelRequest>
    </GetPostageLabel>
  </soap:Body>
</soap:Envelope>
</cfoutput>
</cfsavecontent>

<cfhttp url="https://www.envmgr.com/LabelService/EwsLabelService.asmx?WSDL"
method="POST">
 <cfhttpparam type="HEADER" name="Content-Type" value="text/xml; charset=utf-
8">
 <cfhttpparam type="HEADER" name="Host" value="www.envmgr.com">
```

```
 <cfhttpparam type="HEADER" name="SOAPAction"
value="www.envmgr.com/LabelService/GetPostageLabel">
 <cfhttpparam type="HEADER" name="Content-Length"
value="#len(trim(soap_packet))#">
 <cfhttpparam type="BODY" value="#trim(soap_packet)#">
</cfhttp>

<cfoutput>

<cfset myDoc =
xmlParse(Replacelist(cfhttp.fileContent,"soap:Body,soap:Envelope","body,envel
ope"))>

<cfset Image64 =
#myDoc.envelope.body.GetPostageLabelResponse.LabelRequestResponse.Base64Label
Image.XMLText#>

<cfset ImageData = #toBinary(Image64)#>

<CFSET timestamp =
"#DateFormat(Now(),'yyyymmdd')##TimeFormat(Now(),'HHmmssl')#">

<cffile action="write" file="#ExpandPath('.')#/tempimage#timestamp#.jpg"
output="#ImageData#" attributes="normal">

<img src="tempimage#timestamp#.jpg" />
</cfoutput>
```

C# Examples

This section shows one or more C# (C Sharp) source examples:

- Request a Refund: ConsumeRefundRequest.cs
- Request a List of Transactions: GetTransactionsListing.cs

Request a Refund: ConsumeRefundRequest.cs

The following source example:

- Requests a refund for a label using the Refund Web Service API.
- Uses the C# development environment.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
/*
 * Created by SharpDevelop.
 * User: troy
 * Date: 11/2/2006
 * Time: 11:43 AM
 *
 * To change this template use Tools | Options | Coding | Edit Standard
Headers.
 */
using System;
using System.Xml;
using System.Web.Services;
using System.Collections.Generic;
using System.Web;
using System.Net;
using System.IO;
using System.Text;

namespace ConsumeRefundRequest
{
   class MainClass
   {
      public static void Main(string[] args)
      {
         Console.WriteLine("Start");

         String strAccount = "";
         String strPassPhrase = "";
         String strPICNumber = "";

         String strXML = "<RefundRequest><AccountID>" + strAccount +
"</AccountID><PassPhrase>" + strPassPhrase +
"</PassPhrase><RefundList><PICNumber>" + strPICNumber +
"</PICNumber></RefundList></RefundRequest>";

         string postData = "method=RefundRequest&XMLInput=" + strXML;
```

```
         HttpWebRequest request = (HttpWebRequest)
WebRequest.Create("https://www.endicia.com/ELS/ELSServices.cfc?wsdl");

         request.ContentType = "application/x-www-form-urlencoded";
         request.ContentLength = postData.Length;
         request.Method = "POST";
         request.AllowAutoRedirect = false;

         Stream requestStream = request.GetRequestStream();
         byte[] postBytes = Encoding.ASCII.GetBytes(postData);
         requestStream.Write(postBytes, 0, postBytes.Length);
         requestStream.Close();
         HttpWebResponse response = (HttpWebResponse) request.GetResponse();

         Console.WriteLine(new
StreamReader(response.GetResponseStream()).ReadToEnd());
         Console.WriteLine("Headers:");
         Console.WriteLine(response.Headers.ToString());
      }
   }
}
```

Request a List of Transactions: GetTransactionsListing.cs

The following source example:

- Requests a list of transactions using the Get Transactions Listing API.
- Uses the C# development environment.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
/*
 * Created by SharpDevelop.
 * User: troy
 * Date: 3/6/2007
 * Time: 10:12 AM
 *
 * To change this template use Tools | Options | Coding | Edit Standard
Headers.
 */
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Diagnostics;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Xml;
using System.Xml.Serialization;

namespace GetTransactionsListing
{
   class MainClass
   {
      public static void Main(string[] args)
      {
         string AccountID  =  "";
         string PassPhrase =  "";
         string Connection =  "http://www.endicia.com/";
         string xmlsend    =  "<TransactionsListingRequest><AccountID>" +
AccountID + "</AccountID><PassPhrase>" + PassPhrase + "</PassPhrase>" +

   "<StartDate>02/01/2007</StartDate><StartTime>00:00:00</StartTime><EndDate>
02/07/2007</EndDate><EndTime>23:59:59</EndTime>" +
                           "</TransactionsListingRequest>";

         try
         {
            ELSServicesService request =  new ELSServicesService(Connection);
            System.Xml.XmlNode[] answer   =  {(new
XmlDocument()).FirstChild,(new XmlDocument()).FirstChild};
            XmlDocument send        =  new XmlDocument();
            send.InnerXml           =  xmlsend;
```

```
            answer                    =
   (System.Xml.XmlNode[])request.GetTransactionsListing((object)send.InnerXml
);

            Console.WriteLine(answer[2].OuterXml);
         }
         catch (Exception except)
         {
            Console.WriteLine("Cannot Load XML: " + except.Message + "\n\n" +
xmlsend);
         }
      }
   }
}

[System.CodeDom.Compiler.GeneratedCodeAttribute("wsdl", "1.0.3705.0")]
[System.Diagnostics.DebuggerStepThroughAttribute()]
[System.ComponentModel.DesignerCategoryAttribute("code")]
[System.Web.Services.WebServiceBindingAttribute(Name="ELSServices.cfcSoapBind
ing", Namespace="http://ELS")]
[System.Xml.Serialization.SoapIncludeAttribute(typeof(QueryBean))]
[System.Xml.Serialization.SoapIncludeAttribute(typeof(CFCInvocationException)
)]

public partial class ELSServicesService :
System.Web.Services.Protocols.SoapHttpClientProtocol
{
    public ELSServicesService(string connection)
    {
        this.Url = connection + "/ELS/ELSServices.cfc";
    }

    public ELSServicesService()
    {
        this.Url = "https://www.endicia.com/ELS/ELSServices.cfc";
    }

    [System.Web.Services.Protocols.SoapRpcMethodAttribute("",
RequestNamespace="http://ELS", ResponseNamespace="http://ELS")]
    [return:
System.Xml.Serialization.SoapElementAttribute("GetTransactionsListingReturn")
]

    public object GetTransactionsListing(object XMLInput)
    {
        object[] results = this.Invoke("GetTransactionsListing", new object[]
{XMLInput});
        return ((object)(results[0]));
    }

    public System.IAsyncResult BeginGetTransactionsListing(object XMLInput,
System.AsyncCallback callback, object asyncState)
    {
        return this.BeginInvoke("GetTransactionsListing", new object[]
{XMLInput}, callback, asyncState);
    }
```

```
    public object EndGetTransactionsListing(System.IAsyncResult asyncResult)
    {
        object[] results = this.EndInvoke(asyncResult);
        return ((object)(results[0]));
    }
}

[System.CodeDom.Compiler.GeneratedCodeAttribute("wsdl", "1.0.3705.0")]
[System.SerializableAttribute()]
[System.Diagnostics.DebuggerStepThroughAttribute()]
[System.ComponentModel.DesignerCategoryAttribute("code")]
[System.Xml.Serialization.SoapTypeAttribute(Namespace="http://rpc.xml.coldfus
ion")]

public partial class QueryBean
{
    [System.Xml.Serialization.SoapElementAttribute(IsNullable=true)]

    public string[] columnList;

    [System.Xml.Serialization.SoapElementAttribute(IsNullable=true)]

    public object[] data;
}

[System.CodeDom.Compiler.GeneratedCodeAttribute("wsdl", "1.0.3705.0")]
[System.SerializableAttribute()]
[System.Diagnostics.DebuggerStepThroughAttribute()]
[System.ComponentModel.DesignerCategoryAttribute("code")]
[System.Xml.Serialization.SoapTypeAttribute(Namespace="http://rpc.xml.coldfus
ion")]

public partial class CFCInvocationException
{
}
```

Perl Examples

This section shows one or more Perl source examples:

- Request a First-Class Shipping Label: LabelRequestXML.pl

Request a First-Class Shipping Label: LabelRequestXML.pl

The following source example:

- Requests a First-Class shipping label using the GetPostageLabel Web method of the Endicia Label Server Web Service API.
- Uses Perl.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
#!/usr/bin/perl

use LWP::UserAgent;
use HTTP::Request;
use HTTP::Headers;
use XML::Simple;
use MIME::Base64;

my $objUserAgent = LWP::UserAgent->new;

$request = "<?xml version=\"1.0\" encoding=\"utf-8\"?>
<LabelRequest Test=\"YES\" LabelType=\"Default\" LabelSize=\"4X6\"
ImageFormat=\"GIF\">
 <RequesterID>abcd</RequesterID>
 <AccountID>123456</AccountID>
 <PassPhrase>samplePassPhrase</PassPhrase>
 <MailClass>FIRST</MailClass>
 <DateAdvance>0</DateAdvance>
 <WeightOz>1</WeightOz>
  <Stealth>FALSE</Stealth>
 <Services InsuredMail=\"OFF\" SignatureConfirmation=\"OFF\" />
  <Value>0</Value>
 <Description>Sample Label</Description>
 <PartnerCustomerID>12345ABCD</PartnerCustomerID>
 <PartnerTransactionID>6789EFGH</PartnerTransactionID>
 <ToName>Benjamin Franklin</ToName>
 <ToCompany>United States Postal Service</ToCompany>
 <ToAddress1>475 L'Enfant Plaza SW</ToAddress1>
 <ToCity>Washington</ToCity>
 <ToState>DC</ToState>
 <ToPostalCode>20260</ToPostalCode>
<ToPhone>2025551212</ToPhone>
 <FromName>Technical Support</FromName>
 <FromCompany>DYMO Endicia</FromCompany>
 <ReturnAddress1>385 Sherman Ave</ReturnAddress1>
 <FromCity>Palo Alto</FromCity>
 <FromState>CA</FromState>
 <FromPostalCode>94306</FromPostalCode>
```

```
 <FromZIP4></FromZIP4>
 <FromPhone>8005763279</FromPhone>
</LabelRequest>";

$contentlength = length($request);

$objHeader = HTTP::Headers->new(
 Host           => 'www.envmgr.com',
 Content_Type   => 'application/x-www-form-urlencoded',
 Content_Length => $contentlength
);

$request = "labelRequestXML=$request";

$objRequest = HTTP::Request-
>new("POST","https://www.envmgr.com/LabelService/EwsLabelService.asmx/GetPost
ageLabelXML",$objHeader,$request);

$objResponse = $objUserAgent->request($objRequest);

if (!$objResponse->is_error)
{
  print "sent=$request\n\n";
  $content = $objResponse->content;
  print "respond=$content\n\n";
}
else
{
  $sent = $objRequest->as_string;
  print "$sent\n\n";
  $error = $objResponse->error_as_HTML;
  print "error=$error\n\n";
  print "respond=$content\n\n";
}
```

PHP Examples

This section shows one or more PHP source examples:

- Request a Shipping Label: Endicia.php
- Request a Shipping Label: nuSoapLabelRequest.php
- Request a Shipping Label: PHP5LabelRequest.php
- Request a First-Class Shipping Label: cURLLabelRequest.php
- Request an International Label: PHP5LabelRequestInternationalCustoms.php
- Request a Shipping Rate: PHP5CalculatePostageRate.php
- Request a Refund: curlRefund.php
- Request a Refund Based On PICNumber: formRefund.php
- Request a Refund: nuSoapRefund.php
- Request a Label Refund: php5RefundRequest.php
- Request a New User Signup: nuSoapUserSignup.php
- Request the Status of a Shipment: nuSoapStatus.php
- Request a List of Transactions: GetTransactionsListing API
- Request a Mail Carrier Pickup: nuSoapMLCarrierPickup.php

Request a Shipping Label: Endicia.php

The following source example:

- Retrieves a postage label by opening a URL using fopen() and then making the request using the LabelRequest API.
- Uses PHP.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

$strGetLabelURL =
"https://www.envmgr.com/LabelService/EwsLabelService.asmx/GetPostageLabelXML"
;

$request = '
```

```
<LabelRequest ImageFormat="GIF" Test="YES">
     <RequesterID>abcd</RequesterID>
     <AccountID>123456</AccountID>
     <PassPhrase>samplePassPhrase</PassPhrase>
     <MailClass>FIRST</MailClass>
     <DateAdvance>0</DateAdvance>
     <WeightOz>1</WeightOz>
     <Stealth>FALSE</Stealth>
     <Services InsuredMail="OFF" SignatureConfirmation="OFF" />
     <Value>0</Value>
     <Description>Sample Label</Description>
     <PartnerCustomerID>12345ABCD</PartnerCustomerID>
     <PartnerTransactionID>6789EFGH</PartnerTransactionID>
     <ToName>Benjamin Franklin</ToName>
     <ToCompany>United States Postal Service</ToCompany>
     <ToAddress1>475 L'Enfant Plaza SW</ToAddress1>
     <ToCity>Washington</ToCity>
     <ToState>DC</ToState>
     <ToPostalCode>20260</ToPostalCode>
     <ToPhone>2025551212</ToPhone>
     <FromName>Technical Support</FromName>
     <FromCompany>DYMO Endicia</FromCompany>
     <ReturnAddress1>385 Sherman Ave.</ReturnAddress1>
     <FromCity>Palo Alto</FromCity>
     <FromState>CA</FromState>
     <FromPostalCode>94306</FromPostalCode>
     <FromZIP4>1864</FromZIP4>
     <FromPhone>8005763279</FromPhone>
    </LabelRequest>
';

   $params = array('http' => array(
                  'method' => 'POST',
                  'content' => 'labelRequestXML='.$request,
            'header' => 'Content-Type: application/x-www-form-urlencoded'));

   $ctx = stream_context_create($params);

   $fp = fopen($strGetLabelURL, 'rb', false, $ctx);

      if (!$fp) {
          print "Problem with $strGetLabelURL";
      }
      $response = stream_get_contents($fp);
      if ($response === false) {
          print "Problem reading data from $url, $php_errormsg";
      }
print_r($response);
?>
```

Request a Shipping Label: nuSoapLabelRequest.php

The following source example:

- Uses SOAP to request a shipping label using the Request a Shipping Label method of the Endicia Label Server API.
- Uses PHP4.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

require_once('nusoap.php');

$soapclient = new
soapclient('https://www.envmgr.com/LabelService/EwsLabelService.asmx?wsdl');

$request = "
<GetPostageLabel xmlns=\"www.envmgr.com/LabelService\">
<LabelRequest Test=\"YES\" LabelType=\"Default\" LabelSize=\"4X6\"
ImageFormat=\"GIF\">
 <RequesterID>abcd</RequesterID>
 <AccountID>123456</AccountID>
 <PassPhrase>samplePassPhrase</PassPhrase>
 <MailClass>FIRST</MailClass>
 <DateAdvance>0</DateAdvance>
 <WeightOz>1</WeightOz>
 <Stealth>FALSE</Stealth>
 <Services InsuredMail=\"OFF\" SignatureConfirmation=\"OFF\" />
 <Value>0</Value>
 <Description>Sample Label</Description>
 <PartnerCustomerID>12345ABCD</PartnerCustomerID>
 <PartnerTransactionID>6789EFGH</PartnerTransactionID>
 <ToName>Benjamin Franklin</ToName>
 <ToCompany>United States Postal Service</ToCompany>
 <ToAddress1>475 L'Enfant Plaza SW</ToAddress1>
 <ToCity>Washington</ToCity>
 <ToState>DC</ToState>
 <ToPostalCode>20260</ToPostalCode>
<ToPhone>2025551212</ToPhone>
 <FromName>Technical Support</FromName>
 <FromCompany>DYMO Endicia</FromCompany>
 <ReturnAddress1>385 Sherman Ave</ReturnAddress1>
 <FromCity>Palo Alto</FromCity>
 <FromState>CA</FromState>
 <FromPostalCode>94306</FromPostalCode>
 <FromZIP4>1864</FromZIP4>
 <FromPhone>8005763279</FromPhone>
</LabelRequest>
</GetPostageLabel>
";
```

```
$soapclient-
>call('GetPostageLabel',$request,'http://schemas.xmlsoap.org/soap/envelope/',
'www.envmgr.com/LabelService/GetPostageLabel',false,null,'document','literal'
);

$XMLChar = array(">", "<");
$HTMLChar = array("&gt;","&lt;");

echo '<br /><hr /><br />Request: <pre>';
echo str_replace($XMLChar,$HTMLChar,$soapclient->request);
echo '</pre>';
echo '<br /><hr /><br />Response: <pre>';
echo str_replace($XMLChar,$HTMLChar,$soapclient->response);
echo '</pre>';
?>
```

Request a Shipping Label: PHP5LabelRequest.php

The following source example:

- Uses SOAP to request a shipping label using the Request a Shipping Label method of the Endicia Label Server API.
- Uses PHP5.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

$client = new
SoapClient("https://www.envmgr.com/LabelService/EwsLabelService.asmx?wsdl");

echo "<b>USING ARRAY</b><br /><br />";

$service_type = 'PriorityMailInternational';
$ounces  = '16';
$value = '500.00';
$to_fname = 'Eberhard Anheuser';
$to_company = 'hofbrauhaus';
$to_addr1 = 'Hofbrauallee 1';
$to_city = 'Munich';
$to_state = ' ';
$to_zip5 = '81829';
$to_country = 'Germany';
$to_phone = '6503212640';
$from_fname = 'Bob Smith';
$from_addr1 = '15442 36th Street NW';
$from_city = 'NEWFOLDEN';
$from_state = 'MN';
$from_zip5 = '56738';
$from_zip4 = '';
$from_phone = '2188743305';
$mime =  'GIF';

$data = array
(
   'LabelRequest' => array
   (
      'RequesterID' => 'THEREQUESTER',
      'AccountID' => '123456',
      'PassPhrase' => 'MyPassPhrase',
      'MailClass' => $service_type,
      'DateAdvance' => 0,
      'WeightOz' => $ounces ,
      'CostCenter' => 0,
      'Value' => $value,
      'Services' => array
      (
         'CertifiedMail' => 'OFF',
         'DeliveryConfirmation' => 'OFF',
```

```
            'ElectronicReturnReceipt' => 'OFF',
            'InsuredMail' => 'ENDICIA',
            'SignatureConfirmation' => 'OFF'
        ),
        'Description' => 'Sample Label',
        'PartnerCustomerID' => '12345ABCD',
        'PartnerTransactionID' => '6789EFGH',
        'OriginCountry' => 'United States',
        'ToName' =>  $to_fname,
        'ToCompany' => $to_company,
        'ToAddress1' => $to_addr1,
        'ToCity' => $to_city,
        'ToState' => $to_state,
        'ToPostalCode' => $to_zip5,
        'ToCountry' => $to_country,
        'ToPhone' => $to_phone,
        'FromName' => $from_fname,
        'ReturnAddress1' => $from_addr1,
        'FromCity' => $from_city,
        'FromState' => $from_state,
        'FromPostalCode' => $from_zip5,
        'FromZIP4' => $from_zip4,
        'FromPhone' => $from_phone,
        'CustomsQuantity1' => 0,
        'CustomsValue1' => 0,
        'CustomsWeight1' => 0,
        'CustomsQuantity2' => 0,
        'CustomsValue2' => 0,
        'CustomsWeight2' => 0,
        'CustomsQuantity3' => 0,
        'CustomsValue3' => 0,
        'CustomsWeight3' => 0,
        'CustomsQuantity4' => 0,
        'CustomsValue4' => 0,
        'CustomsWeight4' => 0,
        'CustomsQuantity5' => 0,
        'CustomsValue5' => 0,
        'CustomsWeight5' => 0,
        'Test' => 'YES',
        'LabelSize' => '4x6',
        'LabelType' => 'International',
        'ImageFormat' => $mime,
    )
);

echo "Sending Array:<br />";

print_r($data);
echo "<br /><br />Result Returned:<br />";

//var_dump($client->__getFunctions());

//var_dump($client->__getTypes());

$result = $client->GetPostageLabel($data);

//$image = $result->LabelRequestResponse->Base64LabelImage;
```

```
//header("Content-Type: image/gif");
//print_r(base64_decode($image));

print_r($result);

?>
```

Request a First-Class Shipping Label: cURLLabelRequest.php

The following source example:

- Uses URL to request a First-Class shipping label using the GetPostageLabel Web method of the Endicia Label Server Web Service API.
- Uses PHP4.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

$request = "
<LabelRequest Test=\"YES\" LabelType=\"Default\" LabelSize=\"4X6\"
ImageFormat=\"GIF\">
 <RequesterID>abcd</RequesterID>
 <AccountID>123456</AccountID>
 <PassPhrase>samplePassPhrase</PassPhrase>
 <MailClass>FIRST</MailClass>
 <DateAdvance>0</DateAdvance>
 <WeightOz>1</WeightOz>
 <Stealth>FALSE</Stealth>
 <Services InsuredMail=\"OFF\" SignatureConfirmation=\"OFF\" />
 <Value>0</Value>
 <Description>Sample Label</Description>
 <PartnerCustomerID>12345ABCD</PartnerCustomerID>
 <PartnerTransactionID>6789EFGH</PartnerTransactionID>
 <ToName>Benjamin Franklin</ToName>
 <ToCompany>United States Postal Service</ToCompany>
 <ToAddress1>475 L'Enfant Plaza SW</ToAddress1>
 <ToCity>Washington</ToCity>
 <ToState>DC</ToState>
 <ToPostalCode>20260</ToPostalCode>
<ToPhone>2025551212</ToPhone>
 <FromName>Technical Support</FromName>
 <FromCompany>DYMO Endicia</FromCompany>
 <ReturnAddress1>385 Sherman Ave</ReturnAddress1>
 <FromCity>Palo Alto</FromCity>
 <FromState>CA</FromState>
 <FromPostalCode>94306</FromPostalCode>
 <FromZIP4>1864</FromZIP4>
 <FromPhone>8005763279</FromPhone>
</LabelRequest>";

$postfields = 'labelRequestXML='.$request;

$curl_handle = curl_init ();

curl_setopt ($curl_handle, CURLOPT_URL,
'https://www.envmgr.com/LabelService/EwsLabelService.asmx/GetPostageLabelXML'
);

curl_setopt ($curl_handle, CURLOPT_FOLLOWLOCATION, 1);
```

```
curl_setopt ($curl_handle, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($curl_handle, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt ($curl_handle, CURLOPT_POST, 1);
curl_setopt ($curl_handle, CURLOPT_POSTFIELDS, $postfields);

$curl_result = curl_exec ($curl_handle) or die ("There has been a CURL_EXEC
error");

echo $curl_result;
?>
```

Request an International Label: PHP5LabelRequestInternationalCustoms.php

The following source example:

- Uses SOAP to request an international label using the Request a Shipping Label method of the Endicia Label Server API.
- Uses PHP5.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

$client = new
SoapClient("https://www.envmgr.com/LabelService/EwsLabelService.asmx?wsdl");

//echo "<b>USING ARRAY</b><br /><br />";

$service_type = 'PriorityMailInternational';
$ounces  = '16';
$value = '';
$to_fname = 'Eberhard Anheuser';
$to_company = 'hofbrauhaus';
$to_addr1 = 'Hofbrauallee 1';
$to_city = 'Munich';
$to_state = ' ';
$to_zip5 = '81829';
$to_country = 'Germany';
$to_phone = '6503212640';
$from_fname = 'Bob Smith';
$from_addr1 = '15442 36th Street NW';
$from_city = 'NEWFOLDEN';
$from_state = 'MN';
$from_zip5 = '56738';
$from_zip4 = '';
$from_phone = '2188743305';
$mime   =       'GIF';

$data = array
(
   'LabelRequest' => array
   (
      'RequesterID' => 'THEREQUESTER',
      'AccountID' => '123456',
      'PassPhrase' => 'MyPassPhrase',
      'MailClass' => $service_type,
      'DateAdvance' => 0,
      'WeightOz' => $ounces ,
      'CostCenter' => 0,
      'Value' => $value,
      'Services' => array
      (
         'CertifiedMail' => 'OFF',
         'DeliveryConfirmation' => 'OFF',
```

```
            'ElectronicReturnReceipt' => 'OFF',
            'InsuredMail' => 'OFF',
            'SignatureConfirmation' => 'OFF'
        ),
        'Description' => 'Sample Label',
        'PartnerCustomerID' => '12345ABCD',
        'PartnerTransactionID' => '6789EFGH',
        'OriginCountry' => 'United States',
        'ToName' =>  $to_fname,
        'ToCompany' => $to_company,
        'ToAddress1' => $to_addr1,
        'ToCity' => $to_city,
        'ToState' => $to_state,
        'ToPostalCode' => $to_zip5,
        'ToCountry' => $to_country,
        'ToPhone' => $to_phone,
        'FromName' => $from_fname,
        'ReturnAddress1' => $from_addr1,
        'FromCity' => $from_city,
        'FromState' => $from_state,
        'FromPostalCode' => $from_zip5,
        'FromZIP4' => $from_zip4,
        'FromPhone' => $from_phone,
        'CustomsQuantity1' => 0,
        'CustomsValue1' => 0,
        'CustomsWeight1' => 0,
        'CustomsQuantity2' => 0,
        'CustomsValue2' => 0,
        'CustomsWeight2' => 0,
        'CustomsQuantity3' => 0,
        'CustomsValue3' => 0,
        'CustomsWeight3' => 0,
        'CustomsQuantity4' => 0,
        'CustomsValue4' => 0,
        'CustomsWeight4' => 0,
        'CustomsQuantity5' => 0,
        'CustomsValue5' => 0,
        'CustomsWeight5' => 0,
        'Test' => 'YES',
        'LabelSize' => '4X6',
        'LabelType' => 'International',
        'ImageFormat' => $mime,
    )
);

//echo "Sending Array:<br />";

//print_r($data);
//echo "<br /><br />Result Returned:<br />";

//var_dump($client->__getFunctions());

//var_dump($client->__getTypes());

$result = $client->GetPostageLabel($data);

$image = $result->LabelRequestResponse->Label->Image[0]->_;
```

```
$file = './Temp/Label.gif';

$fh = fopen($file, "w+");
$file_contents = base64_decode($image);
fwrite($fh, $file_contents);
fclose($fh);

$image = $result->LabelRequestResponse->Label->Image[1]->_;

$file = './Temp/Label2.gif';

$fh = fopen($file, "w+");
$file_contents = base64_decode($image);
fwrite($fh, $file_contents);
fclose($fh);

$image = $result->LabelRequestResponse->Label->Image[2]->_;

$file = './Temp/Label3.gif';

$fh = fopen($file, "w+");
$file_contents = base64_decode($image);
fwrite($fh, $file_contents);
fclose($fh);

$image = $result->LabelRequestResponse->Label->Image[3]->_;

$file = './Temp/Label4.gif';

$fh = fopen($file, "w+");
$file_contents = base64_decode($image);
fwrite($fh, $file_contents);
fclose($fh);

//header("Content-Type: image/gif");
//print_r(base64_decode($image));

//print_r($result);

//print_r($image);

?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML lang=en XML:LANG="en" XMLNS="http://www.w3.org/1999/xhtml">
<HEAD>
<meta http-equiv="imagetoolbar" content="no" />
<TITLE>Endicia Label Service - Label.gif</TITLE>
<META http-equiv=Content-Type content="text/html; charset=windows-1252">

</HEAD>
<BODY bgColor=#FFFFFF leftMargin=0 topMargin=0 MARGINWIDTH="0"
MARGINHEIGHT="0">

<TABLE>
<TR>
```

```
<TD><IMG SRC="./Temp/Label.gif" width="50%" height="50%"><br /></TD>
</TR>
<TR>
<TD><IMG SRC="./Temp/Label2.gif" width="50%" height="50%"><br /></TD>
</TR>
<TR>
<TD><IMG SRC="./Temp/Label3.gif" width="50%" height="50%"><br /></TD>
</TR>
<TR>
<TD><IMG SRC="./Temp/Label4.gif" width="50%" height="50%"><br /></TD>
</TR>
</TABLE>

</BODY>
</HTML>
```

Request a Shipping Rate: PHP5CalculatePostageRate.php

The following source example:

- Uses SOAP to request a shipping price using the Calculate Postage and Fees method of the Endicia Label Server API.
- Uses PHP5.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

$client = new
SoapClient("https://www.envmgr.com/LabelService/EwsLabelService.asmx?wsdl");

echo "<b>USING ARRAY</b><br /><br />";

$data = array
(
   'PostageRateRequest' => array
   (
      'RequesterID' => 'Tester1',
      'CertifiedIntermediary' => array
      (
         'AccountID' => '123456',
         'PassPhrase' => 'MyPassPhrase'
      ),
      'MailClass' => 'Priority',
      'WeightOz' => '54',
      'Value' => '100.00',
      'Services' => array
      (
         '@attributes' => array
         (
            'DeliveryConfirmation' => 'TRUE',
            'SignatureConfirmation' => 'OFF'
         )
      ),
      'FromPostalCode' => '94301',
      'ToPostalCode' => '56738'
   )
);

echo "Sending Array:<br />";
print_r($data);
echo "<br /><br />Result Returned:<br />";

$result = $client->CalculatePostageRate($data);

print_r($result);

echo "<br /><br />Using the variables:<br />Status: " . $result-
>PostageRateResponse->Status . "<br />";
```

```
echo "Zone: " . $result->PostageRateResponse->Zone . "<br />";
echo "MailService: " . $result->PostageRateResponse->Postage->MailService .
"<br />";
echo "Rate: " . $result->PostageRateResponse->Postage->Rate . "<br /><br />";

?>
```

Request a Refund: curlRefund.php

The following source example:

- Uses cURL to request a refund for a shipping label using the Refund Web Service API.
- Uses PHP4.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?

$xml = "";
$xml .= "<RefundRequest>\n";
$xml .= "<AccountID></AccountID>\n";
$xml .= "<PassPhrase></PassPhrase>\n";
$xml .= "<RefundList>\n";
$xml .= "<PICNumber>9101148008600086672136</PICNumber>\n";
$xml .= "</RefundList>\n";
$xml .= "</RefundRequest>\n";

$curl_handle = curl_init ();

curl_setopt ($curl_handle, CURLOPT_URL,
'http://www.endicia.com/ELS/ELSServices.cfc?wsdl');

$postfields = array(
   'method'   => 'RefundRequest',
   'XMLInput' => $xml
   );

curl_setopt ($curl_handle, CURLOPT_FOLLOWLOCATION, 1);
curl_setopt ($curl_handle, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($curl_handle, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt ($curl_handle, CURLOPT_POST, 1);
curl_setopt ($curl_handle, CURLOPT_POSTFIELDS, $postfields);

$curl_result = curl_exec ($curl_handle) or die ("There has been a CURL_EXEC
error");

cu
```

Request a Refund Based On PICNumber: formRefund.php

The following source example:

- Uses SOAP to request a refund based on PICNumber using the Refund Web Service API.
- Uses PHP4.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

require_once('nusoap.php');

$soapaddress = 'http://www.endicia.com/ELS/ELSServices.cfc?wsdl';
$soapclient = new soapclient($soapaddress);

$xml = "";
$xml = $xml."<RefundRequest>";
$xml = $xml."<AccountID></AccountID>";
$xml = $xml."<PassPhrase></PassPhrase>";
$xml = $xml."<RefundList>";
$xml = $xml."<PICNumber>9102148008600074274232</PICNumber>";
$xml = $xml."<PICNumber>9102148008600074417219</PICNumber>";
$xml = $xml."</RefundList>";
$xml = $xml."</RefundRequest>";

$params = array(
   'XMLInput' => $xml
      );

/*echo $xml;

$soapclient->call('RefundRequest',$params);

echo 'Response: <xmp>';
echo $soapclient->response;
echo '</xmp>';
*/
?>

<form name="MASTER" method="POST"
action="<?=$soapaddress?>&method=RefundRequest">
<input type="text" value="<?=$xml?>" name="XMLInput">
<input type="submit" name="submit" value="submit">
</form>
```

Request a Refund: nuSoapRefund.php

The following source example:

- Uses SOAP to request a refund using the Refund Web Service API.
- Uses PHP4.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

require_once('nusoap.php');

$soapclient = new
soapclient('http://www.endicia.com/ELS/ELSServices.cfc?wsdl');

$xml = "";
$xml = $xml."<RefundRequest>";
$xml = $xml."<AccountID></AccountID>";
$xml = $xml."<PassPhrase></PassPhrase>";
$xml = $xml."<RefundList>";
$xml = $xml."<PICNumber>9102148008600074274232</PICNumber>";
$xml = $xml."<PICNumber>9102148008600074417219</PICNumber>";
$xml = $xml."</RefundList>";
$xml = $xml."</RefundRequest>";

$params = array(
   'XMLInput' => $xml
      );

echo $xml;

echo($soapclient->call('RefundRequest',$params));

echo 'Response: <xmp>';
echo $soapclient->response;
echo '</xmp>';

?>
```

Request the Status of a Shipment: nuSoapStatus.php

The following source example:

- Uses SOAP to request the status of a shipment using the Status Web Service API.
- Uses PHP4.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

require_once('nusoap.php');

$soapclient = new
soapclient('http://192.168.1.5/ELS/ELSServicesString.cfc?wsdl');

$xml = "";
$xml = $xml."<StatusRequest>";
$xml = $xml."<AccountID></AccountID>";
$xml = $xml."<PassPhrase></PassPhrase>";
$xml = $xml."<StatusList>";
$xml = $xml."<PICNumber>9102148008600074274232</PICNumber>";
$xml = $xml."<PICNumber>9101148008600000041328</PICNumber>";
$xml = $xml."</StatusList>";
$xml = $xml."</StatusRequest>";

$params = array(
   'XMLInput' => $xml
      );

echo $xml;

echo($soapclient->call('StatusRequest',$params));

echo 'Response: <xmp>';
echo $soapclient->response;
echo '</xmp>';

?>
```

Request a List of Transactions: GetTransactionsListing.html

The following source example:

- Requests a list of transactions based on AccountID and PassPhrase using the Get Transactions Listing API.
- Uses JavaScript.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<html>
<head>
<title>POST TEST</title>
<script type="text/javascript">
var AccountID = "";
var PassPhrase = "";

function update(strURL,querystring,areaid)
{
   var xmlHttpReq = false;
    var self = this;

    // Mozilla/Safari
    if (window.XMLHttpRequest)
    {
        self.xmlHttpReq = new XMLHttpRequest();
    }
    // IE
    else if (window.ActiveXObject)
    {
        self.xmlHttpReq = new ActiveXObject("Microsoft.XMLHTTP");
    }
    self.xmlHttpReq.open('POST', strURL + '?' + querystring, true);
    self.xmlHttpReq.setRequestHeader('Content-Type', 'application/x-www-form-
urlencoded');
    self.xmlHttpReq.onreadystatechange = function()
    {
        if (self.xmlHttpReq.readyState == 4)
        {
         var output = self.xmlHttpReq.responseText;
         output = trim(output);
         eval("document.getElementById('" + areaid + "').innerHTML = \"" +
output + "\";");
        }
    }
    self.xmlHttpReq.send(querystring);
}

function trim(thestring)
{
   thestring = thestring.replace(/\n/g, "");
   thestring = thestring.replace(/^\s*|\s*$/g, "");
   thestring = thestring.replace(/\"/g,"'");
```

```
   thestring = thestring.replace(/</g,"&lt;");
   thestring = thestring.replace(/>/g,"&gt;");
   return thestring
}

function UpdateDiv()
{
   xmlsend     =  "<TransactionsListingRequest><AccountID>" + AccountID +
"</AccountID><PassPhrase>" + PassPhrase +
"</PassPhrase><StartDate>02/01/2007</StartDate><StartTime>00:00:00</StartTime
><EndDate>02/07/2007</EndDate><EndTime>23:59:59</EndTime></TransactionsListin
gRequest>";
   querystring =  "&method=GetTransactionsListing&XMLInput=" + xmlsend;

   update("http://www.endicia.com/ELS/ELSServices.cfc?wsdl",querystring,"test
er");
}
</script>

</head>
<body>
   <input type="button" name="button" value="Submit"
onClick="javascript:UpdateDiv();">
   <div id="tester" name="tester" style="height:1024;width:768;border:solid
1px black;"></div>
</body>
</html>
```

Request a Mail Carrier Pickup: nuSoapMLCarrierPickup.php

The following source example:

- Uses SOAP to request a Mail Carrier Pickup using the Carrier Pickup Request API.
- Uses PHP4.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

require_once('nusoap.php');

$soapclient = new
soapclient('http://65.203.54.167/ELS/ELSServices.cfc?wsdl');

$xml = "";
$xml = $xml."<MultiLocationCarrierPickup>";
$xml = $xml."<AccountID></AccountID>";
$xml = $xml."<PassPhrase></PassPhrase>";
$xml = $xml."</MultiLocationCarrierPickup>";

$params = array(
   'XMLInput' => $xml
      );

echo $xml;

echo($soapclient->call('MultiLocationCarrierPickup',$params));

echo '<br>Response: <xmp>';
echo $soapclient->response;
echo '</xmp>';

?>
```

Request a New User Signup: nuSoapUserSignup.php

The following source example:

- Uses SOAP to request a new user signup using the Account Signup API.
- Uses PHP4.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

require_once('nusoap.php');

$soapclient = new
soapclient('http://65.203.54.167/ELS/ELSServices.cfc?wsdl');

$xml = "";
$xml = $xml."<UserSignupRequest>";
$xml = $xml."<CompanyName>Example.COM</CompanyName>";
$xml = $xml."<FirstName>Happy</FirstName>";
$xml = $xml."<MiddleInitial>LS</MiddleInitial>";
$xml = $xml."<LastName>Customer</LastName>";
$xml = $xml."<EmailAddress>someone@endicia.com</EmailAddress>";
$xml = $xml."<EmailConfirm>someone@endicia.com</EmailConfirm>";
$xml = $xml."<PhoneNumber>650-555-1212</PhoneNumber>";
$xml = $xml."<ICertify>Y</ICertify>";
$xml = $xml."<OverrideEmailCheck>Y</OverrideEmailCheck>";
$xml = $xml."<PhysicalAddress>247 High St.</PhysicalAddress>";
$xml = $xml."<PhysicalCity>Palo Alto</PhysicalCity>";
$xml = $xml."<PhysicalState>CA</PhysicalState>";
$xml = $xml."<PhysicalZipCode>94301</PhysicalZipCode>";
$xml = $xml."<PassPhrase>passphrase</PassPhrase>";
$xml = $xml."<ChallengeQuestion>Next door?</ChallengeQuestion>";
$xml = $xml."<ChallengeAnswer>Dentist</ChallengeAnswer>";
$xml = $xml."<PartnerId>xxxx</PartnerId>";
$xml = $xml."<ProductType>LABELSERVER</ProductType>";
$xml = $xml."<CreditCardNumber>4111111111111111</CreditCardNumber>";
$xml = $xml."<CreditCardAddress>247 High St.</CreditCardAddress>";
$xml = $xml."<CreditCardCity>Palo Alto</CreditCardCity>";
$xml = $xml."<CreditCardState>CA</CreditCardState>";
$xml = $xml."<CreditCardZipCode>94301</CreditCardZipCode>";
$xml = $xml."<CreditCardType>V</CreditCardType>";
$xml = $xml."<CreditCardExpMonth>04</CreditCardExpMonth>";
$xml = $xml."<CreditCardExpYear>2013</CreditCardExpYear>";
$xml = $xml."<PaymentType>CC</PaymentType>";
$xml = $xml."</UserSignupRequest>";

$params = array(
   'XMLInput' => $xml
      );

$soapclient->call('UserSignup',$params);
```

```
echo 'Response:<br />';
echo $soapclient->response;
?>
```

Request a Label Refund: php5RefundRequest.php

The following source example:

- Uses SOAP to request a label refund using theRefundRequest API.
- Uses PHP5.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
<?php

$xml = "";
$xml .= "<RefundRequest>";
$xml .= "<AccountID>500000</AccountID>";
$xml .= "<PassPhrase>THETESTPHRASE</PassPhrase>";
$xml .= "<Test>Y</Test>";
$xml .= "<RefundList>";
$xml .= "<PICNumber>9101148008600086672136</PICNumber>";
$xml .= "</RefundList>";
$xml .= "</RefundRequest>";

$client = new SoapClient("https://www.endicia.com/ELS/ELSServices.cfc?wsdl");
//echo "<br /><br />";
//var_dump($client->__getFunctions());
//echo "<br /><br />";
//var_dump($client->__getTypes());

$result = $client->RefundRequest($xml);

echo "<br /><br />";

print_r($result);
?>
```

Ruby Examples

This section shows one or more Ruby source examples:

- Request a Shipping Label: RubyCURLLabelRequest.ruby

Request a Shipping Label: RubyCURLLabelRequest.ruby

The following source example:

- Uses cURL to request a shipping label using the Request a Shipping Label method of the Endicia Label Server API.
- Uses Ruby.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
#! /usr/bin/ruby -rrubygems
xml_request = "
<LabelRequest Test=\"YES\" LabelType=\"Default\" LabelSize=\"4X6\"
ImageFormat=\"GIF\">
 <RequesterID>abcd</RequesterID>
 <AccountID>123456</AccountID>
 <PassPhrase>samplePassPhrase</PassPhrase>
 <MailClass>FIRST</MailClass>
 <DateAdvance>0</DateAdvance>
 <WeightOz>1</WeightOz>
 <Stealth>FALSE</Stealth>
 <Services InsuredMail=\"OFF\" SignatureConfirmation=\"OFF\" />
 <Value>0</Value>
 <Description>Sample Label</Description>
 <PartnerCustomerID>12345ABCD</PartnerCustomerID>
 <PartnerTransactionID>6789EFGH</PartnerTransactionID>
 <ToName>Benjamin Franklin</ToName>
 <ToCompany>United States Postal Service</ToCompany>
 <ToAddress1>475 L'Enfant Plaza SW</ToAddress1>
 <ToCity>Washington</ToCity>
 <ToState>DC</ToState>
 <ToPostalCode>20260</ToPostalCode>
 <ToZIP4>0000</ToZIP4>
 <ToDeliveryPoint>00</ToDeliveryPoint>
 <ToPhone>2025551212</ToPhone>
 <FromName>Technical Support</FromName>
 <FromCompany>DYMO Endicia</FromCompany>
 <ReturnAddress1>385 Sherman Ave</ReturnAddress1>
 <FromCity>Palo Alto</FromCity>
 <FromState>CA</FromState>
 <FromPostalCode>94306</FromPostalCode>
 <FromZIP4>1864</FromZIP4>
 <FromPhone>8005763279</FromPhone>
</LabelRequest>"
```

```
URL =
"https://www.envmgr.com/LabelService/EwsLabelService.asmx/GetPostageLabelXML"

require "curl"
c = Curl::Easy.http_post(URL, Curl::PostField.content('labelRequestXML',
xml_request))
c.follow_location = true
c.ssl_verify_host = false
puts c.body_str
```

VisualBasic.NET Examples

This section shows one or more VisualBasic.NET source examples:

- Requests a Shipping Label: LabelServiceClient.vb

Request a Shipping Label: LabelServiceClient.vb

The following source example:

- Requests a shipping label using the Request a Shipping Label method of the Endicia Label Server API.
- Uses VisualBasic.Net.

Important! Before proceeding, read and understand the **Caution Statements**.

Example

```
Imports LabelServiceClient.EwsLabelServiceProxy

Module LabelServiceClient

    Sub Main()

        Dim ewsLabelService As EwsLabelService
        Dim labelRequest As LabelRequest
        Dim labelRequestResponse As LabelRequestResponse

        labelRequest = New LabelRequest
        labelRequest.Services = New SpecialServices

        labelRequest.Test = "YES"
        labelRequest.ImageFormat = "GIF"
        labelRequest.LabelType = "DEFAULT"
        labelRequest.LabelSize = "4X6"
        labelRequest.RequesterID = "abcd"
        labelRequest.AccountID = "123456"
        labelRequest.PassPhrase = "samplePassPhrase"
        labelRequest.MailClass = "FIRST"
        labelRequest.DateAdvance = "0"
        labelRequest.WeightOz = Convert.ToUInt32(1)
        labelRequest.Stealth = "FALSE"
        labelRequest.Services.InsuredMail = "OFF"
        labelRequest.Value = 0.0
        labelRequest.Description = "Sample Label"
        labelRequest.PartnerCustomerID = "12345ABCD"
        labelRequest.PartnerTransactionID = "6789EFGH"
        labelRequest.ToName = "Benjamin Franklin"
        labelRequest.ToCompany = "United States Postal Service"
        labelRequest.ToAddress1 = "475 L'Enfant Plaza SW"
        labelRequest.ToCity = "Washington"
        labelRequest.ToState = "DC"
        labelRequest.ToPostalCode = "20260"
        labelRequest.ToPhone = "2025551212"
```

```
        labelRequest.FromName = "John Doe"
        labelRequest.ReturnAddress1 = "385 Sherman Ave"
        labelRequest.FromCity = "Palo Alto"
        labelRequest.FromState = "CA"
        labelRequest.FromPostalCode = "94306"
        labelRequest.FromZIP4 = "1864"
        labelRequest.FromPhone = "8005763279"

        ewsLabelService = New EwsLabelService

        labelRequestResponse = ewsLabelService.GetPostageLabel(labelRequest)

        If (labelRequestResponse.Status = 0) Then
            System.Console.WriteLine("GetPostageLabel successful ... Label
image (in base64 string):")
            System.Console.WriteLine(labelRequestResponse.Base64LabelImage)
        Else
            System.Console.WriteLine("GetPostageLabel failed")
            System.Console.WriteLine("Status = " +
labelRequestResponse.Status.ToString())
        End If

    End Sub

End Module
```

Appendix D. Country Code Table

The following table specifies the country code to be used in the ToCountryCode element and the CountryOfOrigin element in an international Label Request.

In some cases, the ISO 3166 standard has identified Country Codes for Territories or Provinces of other countries. If there is a country name in the "Use" column, you may need to confirm which code is expected by the receiving country.

For updated information, see the following sources:

- "Index of Countries and Localities" (from the USPS), which is available at: http://pe.usps.gov/cpim/ftp/manuals/imm/immctry.pdf (April 17, 2011). This is an index of countries that includes links to all the limitations, conditions of mailing, and restrictions for each country.

- "Country names and code elements" (from the ISO), which is available at: http://www.iso.org/iso/country_codes/iso_3166_code_lists/country_names_and_code_elements.htm (through August 9, 2011 updates)

Country Name	Country Code	Use	Includes	Notes
Afghanistan	AF			
Aland Islands	AX	Finland		
Albania	AL			
Algeria	DZ			
Andorra	AD			
Angola	AO			
Anguilla	AI			
Antigua and Barbuda	AG		Redonda	
Argentina	AR			
Armenia	AM			
Aruba	AW			
Australia	AU		Keeling Islands; Lord Howe Island; New South Wales; Queensland; South Australia; Tasmania; Thursday Island; Victoria; Western Australia	
Austria	AT			
Azerbaijan	AZ			
Bahamas	BS			
Bahrain	BH			
Bangladesh	BD			

Country Name	Country Code	Use	Includes	Notes
Barbados	BB			
Belarus	BY			
Belgium	BE			
Belize	BZ		British Honduras	
Benin	BJ		Dahomey	
Bermuda	BM			
Bhutan	BT			
Bolivia	BO			
Bonaire	BQ	Netherlands Antilles		
Bosnia-Herzegovina	BA			
Botswana	BW			
Brazil	BR			
British Virgin Islands	BV		Anegada; Jost Van Dyke; Tortola; Virgin Gorda	
Brunei Darussalam	BN			
Bulgaria	BG			
Burkina Faso	BF			
Burma				
Burundi	BI			
Cambodia	KH		Kampuchea	
Cameroon	CM			
Canada	CA		Alberta; British Columbia; Labrador; Manitoba; New Brunswick; Newfoundland; Northwest Territory; Nova Scotia; Ontario; Prince Edward Island; Quebec; Saskatchewan; Yukon Territory	
Cape Verde	CV			
Cayman Islands	KY			
Central African Republic	CF			
Chad	TD		Tchad	
Chile	CL			
China	CN		Hainan Island; Manchuria; Tibet	
Christmas Island	CX	Australia		
Cocos Island	CC	Australia		
Colombia	CO			
Comoros	KM		Anjouan; Grand Comoro; Moheli	

Country Name	Country Code	Use	Includes	Notes
Congo, Democratic Republic of the	CD			
Congo, Republic of the	CG			
Cook Islands	CK	New Zealand		
Costa Rica	CR			
Cote d'Ivoire	CI		Ivory Coast	
Croatia	HR			
Cuba	CU		Isle of Pines, West Indies	
Curacao	CW	Netherlands Antilles		
Cyprus	CY			
Czech Republic	CZ			
Denmark	DK			
Djibouti	DJ		French Somaliland; Issas	
Dominica	DM			
Dominican Republic	DO			
Ecuador	EC			
Egypt	EG			
El Salvador	SV		Salvador	
Equatorial Guinea	GQ		Annobon Island; Corisco Island; Elobey Islands; Fernando Po; Rio Muni	
Eritrea	ER			
Estonia	EE			
Ethiopia	ET			
Falkland Islands	FK			
Faroe Islands	FO			
Fiji	FJ			
Finland	FI			
France	FR		Corsica	
French Guiana	GF			
French Polynesia	PF		Borabora; French Oceania; Gambier; Hivaoa; Huahine; Marquesas Islands; Moorea; Nukahiva; Raiatea; Rapa; Society Islands; Tahaa; Tahiti; Tuamotou; Tubuai	

Country Name	Country Code	Use	Includes	Notes
French Territory of the Afars and Issas	TF	Djibouti		
Gabon	GA			
Gambia	GM			
Georgia, Republic of	GE			
Germany	DE			
Ghana	GH			
Gibraltar	GI			
Great Britain and Northern Ireland	GB		Alderney; Channel Islands; England; Northern Ireland; Sark; Scotland; United Kingdom; Wales	
Greece	GR		Astypalaia; Chalki; Crete; Dodecanese Islands; Kalymnos; Karpathos; Kassos; Kastellorizon; Kos; Leipsos; Leros; Nissiros; Patmos; Rodos; Symi; Tilos	
Greenland	GL			
Grenada	GD			
Guadeloupe	GP		Barthelemy; Desirade Island; French West Indies; Les Saints Island; Marie Galante; Petite Terre; Saint Bartholomew	
Guatemala	GT			
Guernsey	GG	Channel Islands OR Great Britain and Northern Ireland		
Guinea	GN			
Guinea-Bissau	GW			
Guyana	GY		British Guiana	
Haiti	HT			
Honduras	HN		Swan Islands	
Hong Kong	HK		Kowloon	
Hungary	HU			
Iceland	IS			
India	IN		Andaman Islands; Damao; Diu; Goa; Sikkim	

Country Name	Country Code	Use	Includes	Notes
Indonesia	ID		Borneo; East Timor; Irian Barat; West New Guinea	
Iran	IR		Persia	
Iraq	IQ			
Ireland	IE		Eire	
Isle of Man	IM	Great Britain and Northern Ireland		
Israel	IL			
Italy	IT			
Jamaica	JM			
Japan	JP		Nansil Islands; Okinawa; Shikoku; Tori Shima	
Jersey	JE	Channel Islands OR Great Britain and Northern Ireland		
Jordan	JO		Hashemite Kingdom; Trans-Jordan, Hashemite Kingdom	
Kazakhstan	KZ			
Kenya	KE			
Kiribati	KI		Canton Island; Christmas Island; Enderbury Island; Fanning Island; Gilbert Islands; Ocean Island	
Korea, Democratic People's Republic of	KP	North Korea	North Korea	
Korea, Republic of	KR	South Korea	South Korea	
Kosovo, Republic of	XZ			
Kuwait	KW			
Kyrgyzstan	KG			
Laos	LA			
Latvia	LV			
Lebanon	LB			
Lesotho	LS			
Liberia	LR			
Libya	LY		Cyjrenaica; Fezzan; Tripolitania	

Country Name	Country Code	Use	Includes	Notes
Liechtenstein	LI			
Lithuania	LT			
Luxembourg	LU			
Macao	MO	Macau		
Macedonia, Republic of	MK			
Madagascar	MG	Malagasy Republic	Sainte Marie de Madagascar	
Malawi	MW		Nyasaland	
Malaysia	MY		Johore; Kedah; Kelantan; Labuan; Malacca; Malaya; Negri Sembilan; North Borneo; Pahang; Penang; Perak; Perlis; Province Wellesley; Sabah; Sarawak; Selangor; Trengganu; Wellesley, Province	
Maldives	MV			
Mali	ML			
Malta	MT		Cumino Island; Gozo Island	
Marshall Islands, Republic of the	MH			
Martinique	MQ		French West Indies	
Mauritania	MR			
Mauritius	MU		Rodrigues	
Mayotte	YT	France		
Mexico	MX			
Moldova	MD			
Monaco	MC	France		
Mongolia	MN			
Montenegro	ME			
Montserrat	MS			
Morocco	MA			
Mozambique	MZ			
Myanmar	MM		Burma	
Namibia	NA		South-West Africa	
Nauru	NR			
Nepal	NP			
Netherlands	NL		Holland; Netherlands West Indies; Antilles (Saba; Saint Eustatius)	

Country Name	Country Code	Use	Includes	Notes
New Caledonia	NC		Alofi Island; Huan Island; Isle of Pines; Loyalty Islands	
New Zealand	NZ		Aitutaki, Cook Islands; Atiu, Cook Islands; Avarua; Danger Islands; Hervey, Cook Islands; Manahiki; Palmerston, Avarua; Parry, Cook Islands; Penrhyn, Tongareva; Pukapuka; Rakaanga; Rarotonga, Cook Islands; Savage Island, Niue; Suwarrow Islands; Tongareva	
Nicaragua	NI			
Niger	NE			
Nigeria	NG			
Niue	NU	New Zealand		
Norfolk Island	NF	Australia		
Norway	NO		Spitzbergen	
Oman	OM		Muscat	
Pakistan	PK		Baluchistan	
Panama	PA			
Papua New Guinea	PG		Admiralty Islands; Bismark Archipelago; Bougainville; Buka; New Britain; New Hanover; New Ireland	
Paraguay	PY			
Peru	PE			
Philippines	PH			
Pitcairn Island	PN			
Poland	PL			
Portugal	PT		Azores; Madeira Islands	
Qatar	QA		Doha; Umm Said	
Reunion	RE		Bourbon	
Romania	RO			
Russia	RU			
Rwanda	RW			
Saint Barthelemy	BL	Guadeloupe		
Saint Helena	SH		Ascension; Tristan da Cunha	
Saint Kitts And Nevis	KN		Saint Christopher and Nevis	
Saint Lucia	LC			

Country Name	Country Code	Use	Includes	Notes
Saint Maarten (Dutch)	SX	Netherlands Antilles		
Saint Martin (French)	MF	Guadeloupe		
Saint Pierre and Miquelon	PM		Miquelon	
Saint Vincent and the Grenadines	VC		Grenadines	
San Marino	SM			
Sao Tome and Principe	ST			
Saudi Arabia	SA			
Senegal	SN			
Serbia, Republic of	RS			
Seychelles	SC			
Sierra Leone	SL			
Singapore	SG			
Slovak Republic	SK	Slovakia		
Slovenia	SI			
Solomon Islands	SB		Santa Cruz Islands	
Somalia	SO		Somali Democratic Republic; Somaliland	
South Africa	ZA		Transkei	
South Georgia	GS	Falkland Islands		
Spain	ES		Alhucemas; Balearic Islands; Canary Islands; Ceuta; Chaferinas Islands; Melilla; Penon de Velez de la Gomera; Zafarani Islands	
Sri Lanka	LK		Ceylon	
Sudan	SD			
Suriname	SR			
Swaziland	SZ			
Sweden	SE			
Switzerland	CH			
Syrian Arab Republic	SY	Syria		
Taiwan	TW		Formosa; Penghu Islands; Pescadores Islands; Quemoy	
Tajikistan	TJ			
Tanzania	TZ		Pemba; Zanzibar	
Thailand	TH		Siam	

Country Name	Country Code	Use	Includes	Notes
Timor-Leste	TL	Indonesia		
Togo	TG			
Tokelau	TK	Western Samoa		
Tonga	TO		Friendly Islands	
Trinidad and Tobago	TT		Tobago	
Tunisia	TN			
Turkey	TR			
Turkmenistan	TM			
Turks and Caicos Islands	TC		Caicos Islands	
Tuvalu	TV		Ellice Islands	
Uganda	UG			
Ukraine	UA			
United Arab Emirates	AE		Abu Dhabi; Ajman; Dubai; Fujairah; Ras al Kaimah; Sharja; Trucial States; Umm al Quaiwain	
Uruguay	UY			
Uzbekistan	UZ			
Vanuatu	VU		Banks Island; New Hebrides; Torres Island	
Vatican City	VA			
Venezuela	VE			
Vietnam	VN			
Virgin Islands (British)	VG			
Wallis and Futuna Islands	WF		Futuna	
Samoa	WS		Atafu; Fakaofo; Nukunonu; Savaii Island; Union Group; Upolu Island	
Yemen	YE			
Zambia	ZM			
Zimbabwe	ZW		Rhodesia	

Glossary

Account ID

A unique *Account ID* is assigned by Endicia when you sign up for the Endicia Label Server Web Service. The *Account ID* is associated with an Endicia postage account and holds information about postage transactions as well as available balance for printing postage.

When using the Test Server, you can use any 6 digit number for the Account ID. When on the production server, you will be supplied with an Endicia Label Server Account ID.

Girth

The distance around the thickest part of an object.

IMb and IMpb

Intelligent Mail Barcode and Intelligent Mail Package Barcode, respectively.
These are the latest barcode standards for USPS tracking and will replace the current barcodes used on all classes of mail.

Pass Phrase

A very long and complex password. During the new account sign-up process, you will be asked to select a temporary *Pass Phrase* for the account. A Pass Phrase must be at least 5 characters long with a maximum of 64 characters. For added security, the Pass Phrase should be at least 10 characters long and include more than one word, use at least one uppercase and lowercase letter, one number and one non-text character, such as punctuation.,

Production Server

The *Production Server* contains the live version of the Endicia Label Server Web Service that generates shipping labels with real postage.

Requester ID

A *Requester ID* (also known as Partner ID or Partner Code) uniquely identifies the system making the request.

Test Server

The *Test Server* contains a test version of the Endicia Label Server Web Service that only generates "test" shipping labels with VOID postage. This allows you to develop and test your application's calls to the Endicia Label Server Web Service methods.

Web Password

You use the Web Password to:

- Gain access to the endicia.com Web site.
- View account transactions.
- Change account information.

The Web Password does not enable you to buy and print postage.

Web Service

A *Web Service* is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web Service in a manner prescribed by its description using SOAP-messages,

typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards. Reference source: The W3C at http://www.w3.org/TR/ws-gloss/

INDEX

A

account number
  email message · 136
account status
  flag · 89
  getting · 87
accounts
  Active status · 87
  change Pass Phrases for · 59
  ID · 245
  maintain multiple · 94, 99, 105
  maintain security model of creating · 136
  production, sign up for · 22
  reactivate · 155
  request status of · 87
  sign up · 90, 136
  status flag · 56
AccountStatusRequest
  about · 87
  XML elements, table of · 88
  XML syntax · 87
  XML tags for · 87
  XMLexample of · 88
AccountStatusResponse
  XML elements, table of · 88
  XML syntax · 88
  XML tags for · 88
  XML, example of · 89
Active status, accounts with · 87, 89
address validation
  bypass · 33, 182
  Dial-A-ZIP · 146
  engine, rejection by · 157
addresses
  ambiguous, Dial-A-ZIP · 193
  not found, APO/FPO · 157
  not found, Dial-A-ZIP · 193
  sender's country · 38
  sender's postal code · 38
  unable to validate · 192
  validate all · 33
agreement, Exhibit B · 155
API
  Buying Postage · 54
  Calculate Postage and Fees · 78
  Carrier Pickup · 117
  Carrier Pickup Change · 128
  CarrierPickupRequest · 118
  Changing the Pass Phrase · 59, 62
  Getting Account Status · 87
  Multiple Location Carrier Pickup API · 117
  Requesting a Shipping Label · 23, 90
APO/FPO
  address not found · 157
  customs form type · 41
  customs forms · 157

B

Buying Postage API · 54

C

C Sharp · *See* C#.
C# (C Sharp) examples
  ConsumeRefundRequest.cs · 200
  list of · 200
  Request a Refund · 200
Calculate Postage and Fees API · 78
CalculatePostageRate, to calculate postage · 68, 78
CalculatePostageRateXML, HTTP POST request for · 68, 78
Canceling a Carrier Pickup Request API · 123
carrier pickup
  cancel request · 123
  changing · 127
  PickupList container node · 128
  request a mail · 228
  requesting · 117
  schedule multiple locations · 132
  scheduling · 118
Carrier Pickup API · 117
Carrier Pickup Change API · 128
Carrier Pickup Request · *See also* CarrierPickupRequest.
CarrierPickupCancel
  about · 123
  error codes · 126
  XML elements, table of · 123
  XML, example of · 124
CarrierPickupCancelResponse
  XML elements, table of · 124
  XML syntax · 124
  XML tags for · 124
  XML, example of · 125
CarrierPickupChange
  about · 127
  error codes · 130
  XML elements · 127
  XML, example · 128
CarrierPickupChangeResponse
  XML · 129
  XML elements, table of · 124
  XML elements, table of · 129
  XML syntax · 129
  XML tags · 129
CarrierPickupRequest
  about · 118
  error codes · 122, 135

table of · 118
XML, example of · 120
CarrierPickupResponse
XML elements, table of · 120
XML tags · 120
XML tags for · 132
XML, example of · 120
caution statements, source examples · 196
ChangePassPhraseRequest
about · 59
XML elements, table of · 59, 62, 65
XML syntax · 59, 62, 64
XML tags for · 59, 62, 64
XML, example · 60, 63, 65
ChangePassPhraseRequestResponse
XML elements · 61, 63, 66
XML syntax · 61, 63, 66
XML tags for · 61, 63, 66
XML, example of · 61, 64, 66
changes
history · 161
Changing a Carrier Pickup Request API · 127
Changing the Pass Phrase API · 59, 62
city, invalid, Dial-A-ZIP · 193
Coldfusion
Request a First-Class Shipping Label · 198
source examples · 198
confirmation
delivery · 34, 72, 81
delivery, fee · 51, 76, 84
delivery, not available · 190
email address · 141
number · 140
number for pickup · 124
number of pickup · 133
number to change pickup · 128
number, change or cancel pickup · 117
number, missed pickup · 133
number, pickup · 120
number, to cancel or change pickup · 129
requirements for refunds · 155
signature · 34, 36, 72, 81
signature, fee · 52, 76, 84
signature, not available · 190
ConsumeRefundRequest.cs, source example · 200
Critical Mail · 28, 69
cURLLabelRequest.php, source example · 214
curlRefund.php, source example · 222
Customs Form Options · 151

D

delivery
confirmation · 72, 81
confirmation fee · 51, 76, 84
confirmation not available · 190
delivery confirmation · *See also* confirmation.
Dial-A-ZIP
address not found · 193
address validation · 146
ambiguous address · 193
invalid response · 192
invalid state code · 193
invalid ZIP Code · 193
unknown return code · 193

E

elements
LabelRequest XML syntax · 23
LabelRequestResponse XML syntax · 49, 92
email
account number · 136
address, confirmation · 141
message, Welcome · 94, 99, 105, 117, 136
notifications, customize · 155
obtain support through · 155
send sample label · 155
signed agreement, Exhibit B · 155
Endicia Label Server
Web service · 20
Endicia Label Server Web Service
accessing, through SSL/HTTPS · 21
first-time use · 22
Endicia Printable Postage XML · 146
Endicia.php, source example · 207
EPL2 · 147
error codes
11110 · 44
11111 · 44
11112 · 44
11113 · 44
11114 · 44
11115 · 45
CarrierPickupCancel Error Codes · 126
CarrierPickupChange · 130
CarrierPickupRequest · 122
CarrierPickupRequest error codes · 135
GetTransactionsListing Response Error Codes · 111
list of · 189
RefundRequest · 97
SCANRequest · 115
Status element · 49, 56, 61, 63, 66, 75, 83, 88
StatusRequest · 93, 103
USPS · 120, 122, 124, 126, 129, 131, 135
examples, source
C# · 200
Coldfusion · 198
cURLLabelRequest.php · 214
curlRefund.php · 222
Endicia.php · 207
formRefund.php · 223
GetTransactionsListing · 226
GetTransactionsListing.cs · 202

LabelRequestXML.pl · 205
nuSoapLabelRequest.php · 209
nuSoapMLCarrierPickup.php · 228
nuSoapRefund.php · 224
nuSoapStatus.php · 225
nuSoapUserSignup.php · 229
Perl · 205
PHP · 207
PHP5CalculatePostageRate.php · 220
PHP5LabelRequest.php · 211
PHP5LabelRequestInternationalCustoms.php · 216
php5RefundRequest.php · 231
Request a First-Class Shipping Label · 214
Request a List of Transactions · 202, 226
Request a Refund · 222, 224
Request a Refund Based On PICNumber · 223
Request a Shipping Label · 207, 211, 232, 234
Request a Shipping Rate · 220
Request an International Label · 216
Request the Status of a Shipment · 225
Ruby · 232
RubyCURLLabelRequest.ruby · 232
VisualBasic.NET · 234
Express Mail
counted as Express · 117
International · 69
label limitation · 39
label not available for · 25
mail class · 28, 69
no weekend delivery · 30
phone number, required · 39
recipient's email address · 40
recipient's company name · 39
requirement for pickup change · 131
requirement for pickup request · 122, 135
sender's company name · 38
service level · 30, 71
ship date · 31
shipment time · 31
Sunday and holiday service · 44

F

FAQ · 155
form submissions · 96, 102, 114, 121, 125, 130, 134, 144
formRefund.php, source example · 223
Frequently Asked Questions · 155

G

GetPostageLabel, about · 23, 90
Getting Account Status API · 87
GetTransactionsListing
about · 105
source example · 226
XML elements, table of · 106
XML syntax · 105
XML tags · 105
XML, syntax example · 108
GetTransactionsListing Response
definitions · 111
error codes · 111
XML elements, table of · 108
XML, example · 110
GetTransactionsListing.cs, source example · 202
girth · 245
glossary, basic terms · 245

H

history, changes · 161
holiday service, Express Mail · 44
HTTP POST
request and response · 22
request for CalculatePostageRateXML · 68, 78

I

IMb · 50, 245
international
customs signer, required · 42
label, request for · 216
LabelType element · 24
mail class for · 28
mail, about insurance · 36
mail, sender name · 38
options for shipping · 157
reply postage · 33
Return Receipt not supported · 34

L

LabelRequest
XML elements, table of · 24, 91, 92
XML syntax · 23
XML tags for · 23
XML, example of · 48, 92
LabelRequestResponse
XML elements, table of · 49
XML syntax · 49, 92
XML tags for · 49, 92
XML, example of · 52
LabelRequestXML.pl, source example · 205
LabelServiceClient.vb, source example · 234
LabelServiceInvoke.cfm, source example · 198

M

misprints, requirements for refunds · 155

MultiLocationCarrierPickup
about · 132
API · 117
XML elements · 132
XML, syntax example · 132
MultiLocationCarrierPickupResponse
XML elements, table of · 132
XML syntax · 132
XML, example of · 133

N

notifications, customize email · 155
nuSoapLabelRequest.php, source example · 209
nuSoapMLCarrierPickup.php, source example · 228
nuSoapRefund.php, source example · 224
nuSoapStatus.php, source example · 225
nuSoapUserSignup.php, source example · 229

P

packet size, enforced limits on · 21
Pass Phrases
about · 245
change · 59, 62
reset · 155
temporary, change · 22
Perl examples, source · 205
PHP source examples · 207
PHP5CalculatePostageRate.php, source example · 220
PHP5LabelRequest.php, source example · 211
PHP5LabelRequestInternationalCustoms.php source example · 216
php5RefundRequest.php, source example · 231
pickup
confirmation number · 124, 133
confirmation number for · 124
confirmation number to cancel or change · 117, 120, 129
confirmation number to change · 128
missed, confirrmation number of · 133
PICNumber
request a refund based on · 223
used with status requests · 99
PieceID, used with status requests · 99
placeholders
HTTP POST request and response · 22
HTTP POST request for BuyPostageXML · 54
HTTP POST request for CalculatePostageRateXML · 68, 78
HTTP POST request for ChangePassPhraseXML · 59, 62, 64
HTTP POST request for GetAccountStatusXML · 87
HTTP POST request for GetPostageLabelXML · 23, 90
PostageRateRequest
about · 68, 78
XML elements, table of · 68, 78
XML syntax · 68, 78
XML tags for · 68, 78
XML, example of · 74, 82
PostageRateResponse
XML elements, table of · 75, 83
XML syntax · 75, 83
XML tags for · 75, 83
XML, example of · 77, 84
production account, sign up for · 22
production server · 245
proxy server · 21

Q

questions and answers · 155

R

RDC · 167
reactivate account · 155
RecreditRequest
about · 54
XML elements, table of · 54
XML syntax · 54
XML tags for · 54
XML, example of · 55
RecreditRequestResponse
XML elements, table of · 56
XML syntax · 56
XML tags for · 56
XML, example of · 57
RefundRequest
about · 94
error codes · 97
XML · 95
XML elements,table of · 94
XML syntax · 94
XML tags for · 94
RefundResponse
XML elements, table of · 95
XML syntax · 95
XML tags · 95
XML, example of · 96
refunds
based On PICNumber, source example · 223
requirements for · 155
Request a First-Class Shipping Label
source example · 198
source example · 214
Request a Label Refund, source example · 231
Request a List of Transactions, source example · 202, 226
Request a New User Signup, source example · 229
Request a Refund
Based On PICNumber source example · 223
C# source example · 200

source example · 222, 224
Request a Shipping Label
example of source · 234
source example · 232
source example · 207, 211
Request a Shipping Rate, source example · 220
Request an International Label, source example · 216
Request the Status of a Shipment, source example · 225
Requester ID · 245
Requesting a Carrier Pickup API · 117
Requesting a Refund API · 94
Requesting a Shipping Label, API · 23, 90
Requesting Transaction Lists API · 105
Rubber Stamps
elements · 37
Ruby source example · 232
RubyCURLLabelRequest.ruby, source example · 232

S

sample label
send email message · 155
SCANRequest
about · 112
error codes · 115
XML elements, table of · 112
XML tags for · 112
XML, example · 113
SCANRequest XML syntax · 112
SCANResponse
XML elements, table of · 114
XML syntax · 114
XML tags for · 114
XML, example of · 114
scans
using forms · 112
Scheduling a Carrier Pickup API · 118
Scheduling a Multiple Location Carrier Pickup Request API · 132
server
proxy · 21
signature
confirmation · 34, 72, 81
confirmation fee · 52
confirmation, fee · 76, 84
signature confirmation · *See also* confirmation.
Signing Up for an Account API · 136
SOAP
example of shipping label request · 209
example, request carrier pickup · 228
example, request international label · 216
example, request label refund · 231
example, request new user signup · 229
example, request refund · 223
example, request refund using Refund Web Service · 224
example, request shipment status · 225
example, request shipping label · 211
example, request shipping price · 220
invalid format of request · 195
limits on packet size · 21
submission, example using · 97, 103, 115, 122, 125, 130, 135, 144
what is · 22
solutions, common · 155
source · *See* examples.
examples · 196
source examples
caution statements · 196
Coldfusion · 198
SSL/HTTPS, accessing Endicia Label Server Web Service through · 21
stamps
printable postage · 146
state code
invalid, Dial-A-ZIP · 193
of missed pickup · 133
of pickup · 133
status
accounts with Active · 87
of accounts request · 87
status flag
account · 56
account status, active · 89
status requests · *See also* StatusRequest.
PICNumber or PieceID · 99
submit Web Service · 99
submit, more efficiently · 99
succeeded or failed · 88
to uniquely identify account · 88
StatusCode Definitions · 103
StatusRequest · *See also* status requests.
about · 99
error codes · 93, 103
XML · 100
XML elements, table of · 100, 101
XML syntax · 99
XML, tags for · 99
StatusResponse
XML syntax · 101
XML, tags for · 101
StatusResponse XML, example of · 101
stealth
postage element · 33
Sunday service, Express Mail · 44

T

tables
AccountStatusRequest XML elements · 88
AccountStatusResponse XML elements · 88
Carrier Pickup Request API · 118
CarrierPickupCancel XML elements · 123
CarrierPickupChangeResponse XML elements · 129

CarrierPickupResponse XML elements · 120
ChangePassPhraseRequest XML elements · 59, 62, 65
GetTransactionsListing Response XML elements · 108
GetTransactionsListing XML elements · 106
LabelRequest XML elements · 24, 91, 92
LabelRequestResponse XML elements · 49
MultiLocationCarrierPickupResponse XML elements · 132
PostageRateRequest XML elements · 68, 78
PostageRateResponse XML elements · 75, 83
RecreditRequest XML elements · 54
RecreditRequestResponse XML elements · 56
RefundRequest XML elements · 94
RefundResponse XML elements · 95
SCANRequest XML elements · 112
SCANResponse XML elements · 114
StatusRequest XML elements · 100, 101
UserSignupRequest XML elements · 136
UserSignupResponse XML elements · 140
Test Server · 245
TransactionsListingResponse
XML syntax · 108
XML tags · 108

U

unknown return code, Dial-A-ZIP · 193
UserSignupRequest
about · 136
XML · 139
XML elements, table of · 136
XML syntax · 91, 136
XML tags for · 91, 136
UserSignupResponse
XML elements, table of · 140
XML syntax · 140
XML tags for · 140
XML, example of · 93, 141
Using the SCAN Form API · 112
USPS
error codes · 122, 124, 126, 129, 131, 135

V

validate
address, unable to · 192
all addresses · 33
VisualBasic.NET examples · 234

W

Web Service
automated submission refund requests · 94
Carrier Pickup Request · 228
Endicia Label Server · 20
get transaction listings · 105
Refund · 94
Refund, API · 222, 223, 224
Refund, example of · 200
sign up for accounts · 90, 136
SOAP-based · 22
Status, API · 225
submit pickup requests · 117
submit status requests · 99
use default values · 94

X

XML tags
CarrierPickupChangeResponse XML · 129
CarrierPickupResponse · 120
for AccountStatusRequest · 87
for CarrierPickupCancelResponse · 124
for CarrierPickupChangeResponse · 129
for CarrierPickupResponse · 132
for ChangePassPhraseRequest · 59, 62, 64
for ChangePassPhraseRequestResponse · 61, 63, 66
for LabelRequest · 23
for LabelRequestResponse · 49, 92
for PostageRateRequest · 68, 78
for PostageRateResponse · 75, 83
for RecreditRequest · 54
for RecreditRequestResponse · 56
for SCANRequest · 112
for SCANResponse · 114
for StatusRequest · 99
for UserSignupRequest · 91, 136
for UserSignupResponse · 140
GetTransactionsListing · 105
RefundResponse · 95
StatusResponse · 101
TransactionsListingResponse · 108

Z

ZIP Code
invalid, Dial-A-ZIP · 193
of pickup · 133
ZPLII · 147