

PayPal Express Checkout

Using Alternative Payment Services and the Simple Order API

October 2019

CyberSource[®]
the power of payment

CyberSource Contact Information

For general information about our company, products, and services, go to <http://www.cybersource.com>.

For sales questions about any CyberSource Service, email sales@cybersource.com or call 650-432-7350 or 888-330-2300 (toll free in the United States).

For support information about any CyberSource Service, visit the Support Center: <http://www.cybersource.com/support>

Copyright

© 2019 CyberSource Corporation. All rights reserved. CyberSource Corporation ("CyberSource") furnishes this document and the software described in this document under the applicable agreement between the reader of this document ("You") and CyberSource ("Agreement"). You may use this document and/or software only in accordance with the terms of the Agreement. Except as expressly set forth in the Agreement, the information contained in this document is subject to change without notice and therefore should not be interpreted in any way as a guarantee or warranty by CyberSource. CyberSource assumes no responsibility or liability for any errors that may appear in this document. The copyrighted software that accompanies this document is licensed to You for use only in strict accordance with the Agreement. You should read the Agreement carefully before using the software. Except as permitted by the Agreement, You may not reproduce any part of this document, store this document in a retrieval system, or transmit this document, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written consent of CyberSource.

Restricted Rights Legends

For Government or defense agencies. Use, duplication, or disclosure by the Government or defense agencies is subject to restrictions as set forth the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 and in similar clauses in the FAR and NASA FAR Supplement.

For civilian agencies. Use, reproduction, or disclosure is subject to restrictions set forth in subparagraphs (a) through (d) of the Commercial Computer Software Restricted Rights clause at 52.227-19 and the limitations set forth in CyberSource Corporation's standard commercial agreement for this software. Unpublished rights reserved under the copyright laws of the United States.

Trademarks

Authorize.Net, eCheck.Net, and The Power of Payment are registered trademarks of CyberSource Corporation.

CyberSource, CyberSource Payment Manager, CyberSource Risk Manager, CyberSource Decision Manager, and CyberSource Connect are trademarks and/or service marks of CyberSource Corporation.

All other brands and product names are trademarks or registered trademarks of their respective owners.

Contents

Recent Revisions to This Document 6

About This Guide 7

Audience and Purpose 7

Conventions 7

Notes and Important Statements 7

Text and Command Conventions 8

Related Documents 8

Customer Support 8

Chapter 1 Introduction to PayPal Express Checkout 9

Benefits of Using PayPal Express Checkout 9

Prerequisites 10

PayPal Merchant Set Up 10

PayPal Express Checkout In-Context Flow 11

PayPal Express Order Processing 12

PayPal Express Standard Orders 12

Standard Order Flow 12

PayPal Express Custom Orders 12

Custom Order Flows 13

PayPal Billing Agreements 13

If You Have a Production Account 13

If You Have a Sandbox Account 13

Using PayPal Express Credit 14

Additional PayPal Express Checkout Services 15

Obtaining Transaction Information 16

Dispute and Chargeback Services 16

Chapter 2 Sessions Service 17

Requesting the Sessions Service 17

Chapter 3	Check Status Service	18
	Requesting the Check Status Service	18
	Checking the Status of a Request	19

Chapter 4	Order Service	21
	Requesting the Order Service	21

Chapter 5	Authorization Service	22
------------------	------------------------------	-----------

Chapter 6	Authorization Reversal Service	23
	Requesting the Authorization Reversal Service	23

Chapter 7	Capture Service	24
	Requesting the Capture Service	24

Chapter 8	Sale Service	25
	Requesting the Sale Service	25

Chapter 9	Refund Service	26
	Requesting the Refund Service	26

Chapter 10	Cancel Service	27
	Requesting the Cancel Order Service	27

Chapter 11	Billing Agreements Service	28
	Requesting a Billing Agreement	28

Appendix A	API Fields	29
	Formatting Restrictions	29
	Data Type Definitions	29
	Numbered Elements	30

Working with Multibyte Character Strings	31
Request IDs in Request Messages	32
Request IDs in Reply Messages	33
Request Fields	33
Reply Fields	43

Appendix B	Service Examples	60
	Sessions Service	60
	Order Service	62
	Authorization Service	64
	Authorization Reversal Service	65
	Capture Service	66
	Sale Service	67
	Refund Service	68
	Cancel Service	69
	Billing Agreement Service	70

Appendix C	Check Status Examples	71
	Sessions Service	71
	Order Service	73
	Authorization Service	74
	Capture Service	75
	Sale Service	76
	Refund Service	77
	Billing Agreement Service	78

Appendix D	Reason Codes	79
-------------------	---------------------	-----------

Recent Revisions to This Document

Release	Changes
October 2019	Added updated information about PayPal customer billing addresses and check status service reply fields. See Appendix A, "API Fields," on page 29 .
September 2019	<p>Reorganized document structure:</p> <ul style="list-style-type: none"> ■ Merged content to create a single chapter, Chapter 3, "Check Status Service," on page 18. ■ Moved relevant Simple Order API information to Appendix A, "API Fields," on page 29. ■ Consolidated service examples in Appendix B, "Service Examples," on page 60. ■ Consolidated check status examples in Appendix C, "Check Status Examples," on page 71. <p>Added new field, clientMetadataID in Appendix A, "API Fields," on page 29.</p>
August 2019	Added information about multibyte character strings. See "Working with Multibyte Character Strings," page 31 .
August 2018	Updated the URL to enable In-Context Express Checkout. See "PayPal Express Checkout In-Context Flow," page 11 .
July 2018	Initial public release.
February 2018	This revision contains only editorial changes and no technical updates.

About This Guide

Audience and Purpose

This guide is written for application developers who want to use the CyberSource Simple Order API to integrate PayPal Express Checkout into their order management system.

Implementing PayPal Express Checkout services requires software development skills. You must write code that uses API request and reply fields to integrate PayPal Express Checkout services.

Conventions

Notes and Important Statements



Note

A *Note* contains helpful suggestions or references to material not contained in the document.



Important

An *Important* statement contains information essential to successfully completing a task or learning a concept.

Text and Command Conventions

Convention	Usage
Bold	<ul style="list-style-type: none"> Field and service names in text; for example: Include the ics_applications field. Items that you are instructed to act upon; for example: Click Save.
Screen text	<ul style="list-style-type: none"> XML elements. Code examples and samples. Text that you enter in an API environment; for example: Set the apSessions_run field to <code>true</code>.

Related Documents

- *Getting Started with CyberSource Advanced for the Simple Order API* ([PDF](#) | [HTML](#)) describes how to get started using the Simple Order API.
- [Simple Order API and SOAP Toolkit API Testing Information page](#)

Refer to the Support Center for complete CyberSource technical documentation:

http://www.cybersource.com/support_center/support_documentation

Customer Support

For support information about any CyberSource service, visit the Support Center:

<http://www.cybersource.com/support>

Introduction to PayPal Express Checkout

When you use the PayPal Express Checkout services:

- No additional security key management is required from you.
- You manage your PayPal Express Checkout payments in the CyberSource Business Center as you would other payment types. You can search for the payment and view its details.
- You use the Transaction Details Report and the Transaction Summary Report to manage your PayPal Express Checkout payments.

Benefits of Using PayPal Express Checkout

When you use PayPal Express Checkout, your customer completes the order on your web site rather than on the PayPal web site. You can:

- Receive real-time notification of successful payments.
- Enable your customers to be directed to your web site's final confirmation page.
- Receive notification from CyberSource that the customer's address has been confirmed and that you are eligible for coverage under the PayPal Seller Protection Policy (if you are enrolled). For more information, see "[Dispute and Chargeback Services](#)," page 16.

PayPal requires you to provide three ways for your customers to access PayPal Express Checkout. See [Table 1](#).

Table 1 PayPal Express Checkout Methods

Method	Description
Shortcut	Place the Checkout with PayPal button on your checkout page.
Payment	Place the PayPal button on your payment page or add <i>PayPal</i> as a choice in your payment drop-down list.
Credit	Place the PayPal Credit button on your checkout page.

Prerequisites

Before you process PayPal Express Checkout payments:

- Install a CyberSource client. See:
Getting Started with CyberSource Advanced for the Simple Order API ([PDF](#) | [HTML](#))
- Contact CyberSource Customer Support to set up your CyberSource merchant account. For more information, see "[PayPal Merchant Set Up](#)," page 10.
- If you are using the Simple Order API with XML, make sure that you are using the correct API version for the features that you are implementing. CyberSource recommends that you use the most recent API version.

For general information about the API versions, see [Getting Started with CyberSource Advanced for the Simple Order API](#).

PayPal Merchant Set Up

To process your PayPal Express Checkout payments through CyberSource, you must set up your merchant account by sharing the following information with CyberSource:

- PayPal sandbox account (facilitator email address)
- PayPal client ID (CID)
- PayPal secret
- PayPal merchant ID (MID)

To acquire this information and provide it to CyberSource:

Step 1 If you do not already have one, set up a PayPal business or premier account:

<https://www.paypal.com/us/webapps/mpp/merchant>

Step 2 If you do not already have one, set up a PayPal developer sandbox account:

<https://developer.paypal.com/docs/classic/lifecycle/sandbox/accounts/#create-and-manage-sandbox-accounts>

Step 3 As directed on the PayPal site, get your PayPal credentials by creating a PayPal app:

<https://developer.paypal.com/docs/api/overview/#get-credentials>

Step 4 Save the Sandbox account facilitator email address, the client ID, and the secret for future reference. For example:

- Sandbox account facilitator email address:
merchantuser-facilitator@merchant.com
- Client ID:
AahnQzKLL2vvG_UI6YQy9xct5joMLVoPHW-
1Bv8gCvPkTiNwQSRCvKIKXy8UZZguijBwJTTs_Cjhdz
- Secret:
EOE3eqqeIBY4q8LhsON0-wp2zPb_0SOqPH3sopx_
uwuIMkCug7zw3aKDunstrXmcrGecmpeUJgsqTGO

Step 5 On the PayPal developer site, click **Profile** at the top right corner of the page.

Step 6 Navigate to **My business info** in the left navigation pane. Your merchant ID is displayed in the list of merchant information. Save the merchant ID for future reference. For example:

- Merchant ID: PP45KFYYH69KU

Step 7 Contact CyberSource merchant support and provide them with the information you have collected from PayPal: the sandbox facilitator email account, the client ID, the secret, and the merchant ID.

When your CyberSource account is set up, you can process PayPal Express Checkout payments.

PayPal Express Checkout In-Context Flow



To enable PayPal Express Checkout, see:

<https://developer.paypal.com/docs/checkout/how-to/server-integration/>

When your customer is ready to pay for the products that they choose from your web site:

- 1 The customer clicks **Checkout with PayPal** on your checkout page or clicks **PayPal** on your payment page.
- 2 Your web site remains in view while a PayPal window appears.
- 3 The customer logs in and chooses a payment method and shipping address.

- 4 Customer confirms payment, and PayPal redirects the customer's browser to your web site. You can choose to display a payment confirmation page:

<https://developer.paypal.com/docs/integration/direct/express-checkout/integration-jsv4/script-options/#show-a-confirmation-page>

PayPal Express Order Processing

PayPal Express checkout provides two kinds of order processing:

- Standard orders
- Custom orders

The kind of order processing you use depends upon your business. CyberSource recommends using custom orders, because they can be more flexible than standard orders.

PayPal Express Standard Orders

Use a standard order when you meet one or more of these criteria:

- You do not usually run out of products.
- You do not usually have back orders.
- You do not accept advanced orders.

Standard Order Flow

Sequence of service requests for a standard 100.00 order:

- 1 Sessions Service 100.00
- 2 Check Status Service
- 3 Order Service for 100.00
- 4 Sale Service for 100.00

PayPal Express Custom Orders

A PayPal Express checkout custom order enables you to perform multiple authorizations and multiple captures for each authorization. Use a custom order when you meet one or more of these criteria:

- You often have split shipments.

- You accept advanced orders for products that are not currently available but will be available within the next 29 days.
- You often have back orders for which the customer waits longer than a week.
- You often reauthorize bank cards.

Custom Order Flows

Table 2 Sequence of Service Requests for Custom 100.00 Orders

Immediate Sale	Delayed Authorization	Delayed Capture
1 Sessions Service	1 Sessions Service	1 Sessions Service
2 Check Status Service	2 Check Status Service	2 Check Status Service
3 Order Service for 100.00	3 Order Service for 100.00	3 Order Service for 100.00
4 Sale Service for 100.00	4 Authorization Service for 70.00	4 Authorization Service for 70.00
	5 Capture Service for 70.00	5 Capture Service for 20.00
	6 Authorization Service for 30.00	6 Capture Service for 50.00
	7 Capture Service for 30.00	7 Authorization Service for 30.00
		8 Capture Service for 15.00
		9 Capture Service for 15.00

PayPal Billing Agreements

You can use a PayPal Express Checkout billing agreement to bill the customer without requiring an authorization for each payment. When you send a request to bill the customer, use the billing agreement to process the PayPal Express checkout order. A billing agreement is also referred to as a *reference transaction*.

If You Have a Production Account

Contact your PayPal account manager to configure your production account for processing billing agreements, or contact PayPal Business Support.

If You Have a Sandbox Account

If your PayPal sandbox account was created before December 15, 2015, contact your PayPal account manager or [PayPal merchant technical support](#) to configure your account to process billing agreements.

If your PayPal sandbox account was created on or after December 15, 2015, it is already configured to process billing agreements.

To request a billing agreement with payment:

- Step 1** Request the sessions service to receive the redirection URL that directs the customer to PayPal to confirm their billing agreement. See [Chapter 2, "Sessions Service," on page 17](#). No amount or currency value is required in the sessions service request when you create a billing agreement.
 - Step 2** Request the billing agreement service and include the request ID field returned in the sessions service response. See [Chapter 11, "Billing Agreements Service," on page 28](#).
 - Step 3** Request the check status service to receive customer details from PayPal for seller protection and other uses. This step is optional. See [Chapter 3, "Check Status Service," on page 18](#).
 - Step 4** Request the sale service to process the customer's billing agreement. Use the billing agreement created in [Step 2](#). See [Chapter 8, "Sale Service," on page 25](#).
-

Using PayPal Express Credit



Note

PayPal Express Credit is supported only in the U.S.A.

The PayPal Express Credit button on your checkout page offers PayPal Express credit as a standalone option. Customers can defer payment by using the PayPal Express credit line included in their PayPal account. Do not display the PayPal Credit button on your shopping cart page if you require customers to log in before checking out.



Important

The PayPal Express credit supports standard order flow (see ["PayPal Express Standard Orders," page 12](#)) and custom order flows (see ["PayPal Express Custom Orders," page 12](#)). You must include the `apSessionsService_paymentOptionID` request field in the sessions request and set it to `Credit`.

When your customers are ready to pay for your product, they follow these steps:

- 1** Click **PayPal Credit** on the checkout page.
- 2** Log in and enter a date of birth and Social Security number.

- 3 Agree to the PayPal credit terms and conditions.
- 4 Confirm the email address and credit line.
- 5 Confirm the credit amount. The customer is directed to the merchant web site.
- 6 Confirm order.

To add the PayPal Express Credit banner and button to your page:

Step 1 Add the PayPal banner to advertise and explain the benefits of PayPal credit:

https://financing.paypal.com/docs/Web_Step_By_Step_Guide.pdf

Step 2 Add the PayPal Credit button to your checkout page:

<https://www.paypal.com/us/webapps/mpp/logos-buttons>

Additional PayPal Express Checkout Services

Table 3 describes additional available services.

Table 3 Additional PayPal Express Checkout Services

Service	Description
Authorization reversal service (see page 23)	<p>Reverses an authorization or an order setup:</p> <ul style="list-style-type: none"> ■ If you performed multiple authorizations for an order and want to reverse one of them, you must specify which authorization to reverse by including the apAuthReversalService_authRequestID field.
Refund service (see page 26)	<p>Refunds the capture amount to a customer. You can perform multiple partial refunds for each capture:</p> <ul style="list-style-type: none"> ■ To refund an immediate full or partial sale, request the refund service and include the apRefundService_refundRequestID field. ■ To refund a delayed capture, you must specify which capture to credit by including the apRefundService_refundRequestID field.

Obtaining Transaction Information

The following table describes how to obtain information about your PayPal Express Checkout transactions.

Table 4 Obtaining Transaction Information

Method	Description
PayPal Express Checkout services	Use the check status service (see Chapter 3, "Check Status Service," on page 18).
Reply messages	After you send a request message for a PayPal Express Checkout service, CyberSource responds with a reply message that contains information about the status of your request, including errors and exceptions.
Reports	Use the Transaction Details Report and the Transaction Summary Report to manage your PayPal Express Checkout payments.
Transaction details	View the details of your PayPal Express Checkout transactions in the Business Center just as you can for other payment types. You can search for transactions by date, application type, customer name, and other transaction identifiers.

Dispute and Chargeback Services

The following table describes PayPal Express Checkout services related to disputes and chargebacks.

Table 5 Services for Disputes and Chargebacks

Service	Description
Chargeback dispute rights	When your customers choose a payment card as the funding source for the PayPal Express Checkout payment, they have chargeback dispute rights. If your customers dispute a charge, PayPal Express Checkout performs the initial chargeback processing and contacts you for documentation.
Buyer complaint process	PayPal Express Checkout offers a Buyer Complaint Process that applies to all purchases regardless of the funding source. When a customer files a complaint, PayPal investigates and contacts you for documentation. See http://www.paypal.com/cgi-bin/webscr?cmd=p/gen/buyer-complaint-outside .

Sessions Service

Requesting the Sessions Service

The **apSessionsService** service:

- Creates a payment or billing agreement with PayPal, which sets up the customer order. See "[Billing Agreements Service](#)," page 28.
- Returns a CyberSource request ID (**requestID**) and a PayPal transaction ID (**apSessionsReply_processorTransactionID**).
- Returns a redirect URL (**apSessionsReply_merchantURL**) that includes a sessions token. Use this redirect URL to direct the customer's browser to the PayPal web site. For example:

```
https://www.sandbox.paypal.com/cgi-bin/webscr?cmd=_express-checkout&token=EC-2HF65459FN270362N
```

For more information:

<https://developer.paypal.com/docs/integration/direct/express-checkout/integration-jsv4/>

For detailed descriptions of required and optional fields, see [Appendix A, "API Fields,"](#) on page 29.

Check Status Service

The check status service is sent to check the status of an individual service request. For example, when your customer initiates a purchase, you send a sessions service request. You can then check the status of that sessions service request using the check status service (**apCheckStatusService**).

The check status service is used to:

- Check the status of individual service requests. For a list of the services you can status check, see ["Checking the Status of a Request," page 19](#).
- Get the details of a billing agreement.



Note

To get the details of a customer's billing address from PayPal, you must have this feature enabled on your PayPal account. Contact your PayPal account manager.

Requesting the Check Status Service

Using the check status service:

- Requires the request ID value that was returned in the individual service reply.
- Returns information about the status of the requested service.

If you are using the check status service to create a billing agreement:

- Requires the billing agreement ID.
- Returns the billing agreement details.



Note

If your **apSessionsService** request includes **item_#** fields, the **apCheckStatusService** service reply returns **item_#** fields. To configure your account for **item_#** fields, contact CyberSource customer support.

For detailed descriptions of required and optional fields, see [Appendix A, "API Fields," on page 29](#).

Checking the Status of a Request

To check the status of a specific service, you need to include the correct request ID. [Table 6](#) describes the request IDs used to check the status of a service.

For more information on the request IDs used by each PayPal Express service, see ["Request IDs in Request Messages," page 32](#).

To check the status of an individual service request:

- Step 1** Set the `apCheckStatusService_run` field to `true`.
- Step 2** Include the following **required** fields in the request:
- `merchantID`
 - `merchantReferenceCode`
 - `apPaymentType`—set to `PPL`.
- Step 1** Set the `requestID` field based on the values described in [Table 6](#).

Table 6 Checking the Status of Individual Services

Service	Request ID Value
Authorization	<p>To check the status of an authorization service request, set the <code>apCheckStatusService_checkStatusRequestID</code> field to the value returned in the <code>requestID</code> field of the authorization service reply. See Example 24, "Check Status Request: Authorization Service," on page 74.</p> <p>The <code>apCheckStatusReply_paymentStatus</code> field returns one of the following values:</p> <ul style="list-style-type: none"> ■ <code>AUTHORIZED</code> ■ <code>EXPIRED</code> ■ <code>FAILED</code> ■ <code>PENDING</code>
Billing Agreement	<p>To check the status of a billing agreement service request, set the <code>apCheckStatusService_sessionsRequestID</code> field to the value returned in the <code>requestID</code> field of the sessions service reply.</p> <p>Also in the billing agreement service check status request, set the <code>ap_billingAgreementID</code> field to the value returned in the <code>apReply_billingAgreementID</code> field in the billing agreement service reply. See Example 32, "Check Status Request: Billing Agreement Service," on page 78.</p> <p>The <code>apCheckStatusReply_paymentStatus</code> field returns one of the following values:</p> <ul style="list-style-type: none"> ■ <code>ACTIVE</code> ■ <code>CANCELLED</code> ■ <code>FAILED</code>

Table 6 Checking the Status of Individual Services (Continued)

Service	Request ID Value
Capture	<p>To check the status of a capture service request, set the apCheckStatusService_checkStatusRequestID field to the value returned in the requestID field of the capture service reply. See Example 26, "Check Status Request: Capture Service," on page 75</p> <p>The apCheckStatusReply_paymentStatus field returns one of the following values:</p> <ul style="list-style-type: none"> ■ DISPUTED ■ FAILED ■ PENDING ■ SETTLED
Order	<p>To check the status of an order service request, set the apCheckStatusService_checkStatusRequestID field to the value returned in the requestID field of the order service reply. See Example 22, "Check Status Request: Order Service," on page 73.</p> <p>The apCheckStatusReply_paymentStatus field returns one of the following values:</p> <ul style="list-style-type: none"> ■ CANCELLED ■ CREATED ■ FAILED
Refund	<p>To check the status of a refund service request, set the apCheckStatusService_checkStatusRequestID field to the value returned in the requestID field of the refund service reply. See Example 30, "Check Status Request: Refund Service," on page 77.</p> <p>The apCheckStatusReply_paymentStatus field returns one of the following values:</p> <ul style="list-style-type: none"> ■ FAILED ■ PENDING ■ REFUNDED
Sale	<p>To check the status of a sale service request, set the apCheckStatusService_checkStatusRequestID field to the value returned in the requestID field of the sale service reply. See Example 28, "Check Status Request: Sale Service," on page 76.</p> <p>The apCheckStatusReply_paymentStatus field returns one of the following values:</p> <ul style="list-style-type: none"> ■ DISPUTED ■ FAILED ■ PENDING ■ SETTLED
Sessions	<p>To check the status of a sessions service request, set the apCheckStatusService_sessionsRequestID field to the value returned in the requestID field of the sessions service reply. See Example 20, "Check Status Request: Sessions Service," on page 71.</p> <p>The apCheckStatusReply_paymentStatus field returns one of these status values:</p> <ul style="list-style-type: none"> ■ CREATED ■ FAILED

Order Service

Requesting the Order Service

The order service:

- Requires the request ID value that was returned in the sessions response.
- Requires the payer ID value to complete the payment.
- Creates an order in anticipation of one or more authorizations.
- Returns a transaction ID.



Important

When any of the additional amount fields change, all of the amount fields should be included in the new request, and they must equal the grand total amount figure. The subtotal amount cannot be updated after a sessions request.



Note

When you are processing a standard order, the next service request is the sale service request. See "[Sale Service](#)," [page 25](#). When you are processing a custom order, the next service request is an authorization service request. See "[Authorization Service](#)," [page 22](#)

For detailed descriptions of the required and optional fields, see [Appendix A, "API Fields,"](#) on [page 29](#).

Authorization Service

The authorization service:

- Requires the request ID value that was returned in the order response.
- Obtains the authorization.
- Enables you to perform multiple partial authorizations.

The authorization service typically commits funds for three business days. If required, contact PayPal to increase the honor interval to more than three days. You can try to capture the authorization for a period of three days after the authorization, and if funds are available, the capture is successful.



If you are processing a custom order, the next service request is a capture service. See ["Capture Service," page 24](#).

For detailed descriptions of required and optional fields, see [Appendix A, "API Fields," on page 29](#).

Authorization Reversal Service

Requesting the Authorization Reversal Service

**Note**

Reversing an authorization that has been captured has no effect on the captured amount.

The authorization reversal service releases the hold on a customer's funds and requires the request ID value that was returned in the authorization response.

For detailed descriptions of required and optional fields, see [Appendix A, "API Fields,"](#) on [page 29](#).

Capture Service

Requesting the Capture Service

**Note**

A capture transfers the funds from the customer's account to your account.

The capture service:

- Requires the request ID value that was returned in the authorization response.
- Enables you to capture the entire authorized amount.
- Enables you to perform multiple partial captures, which are also called *split shipments*, of up to 115% of the authorized amount.

For detailed descriptions of required and optional fields, see [Appendix A, "API Fields,"](#) on [page 29](#).

Requesting the Sale Service



Note

When there is a single capture for the order, you must request the sale service. For multiple captures, you must request the authorization and capture services.

The sale service:

- Requires the request ID value returned in the order response. Include this value in the **apSaleService_orderRequestID** field.
- Requires the billing agreement ID value, if you are processing a billing agreement. Include this value in the **ap_billingAgreementID** field. See "[Billing Agreements Service](#)," page 28.
- Obtains the authorization.
- Captures the authorized amount.
- Returns a transaction ID.

For detailed descriptions of required and optional fields, see [Appendix A, "API Fields,"](#) on page 29.

Refund Service

Requesting the Refund Service

The refund service:

- Requires the request ID value that was returned in the capture response or the sale response. Include this value in the **apRefundService_refundRequestID** field.
- Enables you to perform a full refund or multiple partial refunds for an order.

For detailed descriptions of required and optional fields, see [Appendix A, "API Fields,"](#) on [page 29](#).

Cancel Service

Requesting the Cancel Order Service



Important

You cannot cancel an order that has been authorized or captured.

The cancel service:

- Voids or cancels the order.
- Requires the request ID value that was returned in the order response. Include the value in the **apCancelService_orderRequestID** field.
- When processing a billing agreement transaction, include the **ap_billingAgreementID** field. See ["Billing Agreements Service," page 28](#).

For detailed descriptions of required and optional fields, see [Appendix A, "API Fields," on page 29](#).

Billing Agreements Service

**Important**

Your merchant account must be configured to process billing agreement transactions. See ["PayPal Billing Agreements," page 13](#).

Requesting a Billing Agreement

A billing agreement is set up between PayPal and your customer. When you log the billing agreement, you can easily bill your customer without gathering customer information every time.

The following sequence describes the services you use for a billing agreement.

- 1 Sessions service request—the service reply includes a merchant URL that redirects the customer to your website, where they confirm their billing information. See [Chapter 2, "Sessions Service," on page 17](#).
- 2 Billing agreement service request—creates the billing agreement. Requires the request ID value returned in the sessions service reply.
- 3 Check status service request—requires the request ID returned in the sessions reply and the billing agreement ID returned in the billing agreement reply. See [Chapter 3, "Check Status Service," on page 18](#).
- 4 Sale service—requires the billing agreement ID returned in the billing agreement response. See [Chapter 8, "Sale Service," on page 25](#).
- 5 Refund service—requires the request ID value returned in the sale response. See [Chapter 9, "Refund Service," on page 26](#).
- 6 Cancel service—requires the billing agreement ID returned in the billing agreement response. (See [Chapter 10, "Cancel Service," on page 27](#).)

For detailed descriptions of required and optional fields, see [Appendix A, "API Fields," on page 29](#).

API Fields

Formatting Restrictions

Unless otherwise noted, all field names are case sensitive, and all fields accept special characters such as @, #, and %.



Note

The values of the **item_#_** fields must not contain carets (^) or colons (:) because these characters are reserved for use by the CyberSource services.

Values for request-level and item-level fields must not contain new lines or carriage returns. However, they can contain embedded spaces and any other printable characters. CyberSource removes all leading and trailing spaces.

Data Type Definitions

For more information about these data types, see the [World Wide Web Consortium \(W3C\) XML Schema Part 2: Datatypes Second Edition](#).

Table 7 Data Type Definitions

Data Type	Description
Date and time	Format is YYYY-MM-DDThh:mm:ssZ, where: <ul style="list-style-type: none"> ■ T separates the date and the time ■ Z indicates Coordinated Universal Time (UTC), also known as Greenwich Mean Time (GMT) <p>Example 2019-08-11T22:47:57Z equals August 11, 2019, at 22:47:57 (10:47:57 p.m.).</p>
Integer	Whole number {..., -3, -2, -1, 0, 1, 2, 3, ...}
String	Sequence of letters, numbers, spaces, and special characters

Numbered Elements

The CyberSource XML schema includes several numbered elements. You can include these complex elements more than once in a request. For example, when a customer order includes more than one item, you must include multiple `<item>` elements in your request. Each item is numbered, starting with 0. The XML schema uses an `id` attribute in the item's opening tag to indicate the number. For example:

```
<item id="0">
```

As a name-value pair field name, this tag is called **item_0**. In this portion of the field name, the underscore before the number does not indicate hierarchy in the XML schema. The item fields are generically referred to as **item_#_<element name>** in the documentation.

Below is an example of the numbered `<item>` element and the corresponding name-value pair field names. If you are using the Simple Object Access Protocol (SOAP), the client contains a corresponding `Item` class.

Example 1 Numbered XML Schema Element Names and Name-Value Pair Field Names

XML Schema Element Names	Corresponding Name-Value Pair Field Names
<pre><item id="0"> <unitPrice> <quantity> </item></pre>	<pre>item_0_unitPrice item_0_quantity</pre>
<pre><item id="1"> <unitPrice> <quantity> </item></pre>	<pre>item_1_unitPrice item_1_quantity</pre>



When a request is in XML format and includes an `<item>` element, the element must include an `id` attribute. For example: `<item id="0">`.

Working with Multibyte Character Strings

Some languages—such as Korean, Chinese, and Russian—require more than one ASCII byte per character. Therefore, a string of multibyte characters can exceed the allotted string length of some CyberSource API fields.

In some languages, a short name that visually has only three or four characters might be equivalent to 10 or 15 bytes in UTF-8. Therefore, it is important to consider the overall length of a multibyte string to be sure it fits into the allotted ASCII string length.



For best success when working with languages, make sure the total length of a multibyte string does not exceed the ASCII string length for a specific API field.

Request IDs in Request Messages

The following table lists the request ID field names used in request messages. For the request ID field names used to check the status of an individual service, see "[Checking the Status of a Request](#)," page 19.

Table 8 Request IDs in Request Messages

Service Request	Request Field Names
Check Status (get pay details)	<ul style="list-style-type: none"> ■ apCheckStatusService_sessionsRequestID—set to the value of the requestID field returned in the sessions service response. ■ apCheckStatusService_checkStatusRequestID—set to the value of the requestID field returned in these responses: <ul style="list-style-type: none"> • Order service response • Authorization service response • Capture service response • Sale service response • Refund service response • Billing agreement service response
Billing Agreement (reference transaction)	<ul style="list-style-type: none"> ■ apBillingAgreementService_sessionsRequestID—set to the value of the requestID field returned in the sessions service response. ■ apCheckStatusService_checkStatusRequestID—set to the value of the requestID field returned in the billing agreement service response.
Order	<ul style="list-style-type: none"> ■ apOrderService_sessionsRequestID—set to the value of the requestID field returned in the sessions service response. ■ apCheckStatusService_checkStatusRequestID—set to the value of the requestID field returned in the order service response.
Sale	<p>apSaleService_orderRequestID—set to the value of the requestID field returned in the order service response.</p> <p>or</p> <p>ap_billingAgreementID—include this field if processing a billing agreement.</p>
Authorization	apAuthService_orderRequestID —set to the value of the requestID field returned in the order service response.
Capture	apCaptureService_authRequestID —set to the value of the requestID field returned in the authorization service response.
Refund	<ul style="list-style-type: none"> ■ apRefundService_refundRequestID—set to the value of the requestID field returned in the capture service response. ■ apRefundService_refundRequestID—set to the value of the requestID field returned in the sale service response.

Table 8 Request IDs in Request Messages (Continued)

Service Request	Request Field Names
Authorization Reversal	apAuthReversalService_authRequestID —set to the value of the requestID field returned in the authorization service response.
Cancel	<ul style="list-style-type: none"> ■ apCancelService_orderRequestID—set to the value of the requestID field returned in the order service response. ■ ap_billingAgreementID—set to the value returned in the billing agreement service response.

Request IDs in Reply Messages

For all PayPal Express Checkout services, the request ID is returned in the reply message in the **requestID** field.

Request Fields

Table 9 Request Fields

Field	Description	Required (R) / Optional (O)	Data Type & Length
apAuthReversalService_authRequestID	Set to the value of the requestID field returned in the authorization service response.	Auth Reversal (R)	String (26)
apAuthService_orderRequestID	Set to the value of the requestID field returned in the order service response.	Authorization (R)	String (26)
apAuthService_run	The authorization service request name.	Authorization (R)	String (26)
ap_billingAgreementDescription	Description of the billing agreement.	Sessions (O for billing agreement)	String (127)
ap_billingAgreementID	The identifier for the billing agreement. Set to the value returned in the apReply_billingAgreementID field.	Cancel (O) Check Status (R when checking the status of a billing agreement) Sale (R for a reference transaction)	String (50)

Table 9 Request Fields (Continued)

Field	Description	Required (R) / Optional (O)	Data Type & Length
ap_billingAgreementIndicator	Indicates whether the transaction is a billing agreement. Possible values: <ul style="list-style-type: none"> ■ true ■ false (default) 	Billing Agreement (R)	String (5)
apBillingAgreementService_run	The billing agreement service request name.	Billing Agreement (O)	String (127)
apBillingAgreementService_sessionsRequestID	Set to the value of the requestID field returned in the sessions service response.	Billing Agreement (R)	String (26)
apCancelService_orderRequestID	Set to the value of the requestID field returned in the order service response.	Cancel (R)	String (26)
apCancelService_run	The cancel service request name.	Cancel (R)	String (26)
apCaptureService_isFinal	Indicates whether to release the authorization hold on the remaining funds. Possible values: <ul style="list-style-type: none"> ■ true ■ false (default) 	Capture (O)	String (5)
apCaptureService_run	The capture service request name.	Capture (O)	String (26)
apCheckStatusService_run	The check status service request name.	Check Status (R)	String (26)
apCheckStatusService_sessionsRequestID	Set to the value of the requestID field returned in the sessions service response.	Check Status (R)	String (26)
ap_fundingSource	Payment method for the unit purchase. Possible values: <ul style="list-style-type: none"> ■ UNRESTRICTED (default)—this value is available only when configured by PayPal for the merchant. ■ INSTANT 	Refund (O) Sale (O)	String (30)
ap_payerID	The ID of the customer.	Order (R)	String(30)

Table 9 Request Fields (Continued)

Field	Description	Required (R) / Optional (O)	Data Type & Length
apPaymentType	Identifier for the payment type. Value: PPL	Authorization (R) Auth Reversal (R) Billing Agreement (R) Cancel (R) Capture (R) Check Status (R) Order (R) Refund (R) Sale (R) Sessions (R)	String (3)
apOrderService_run	The order service request name.	Order (R)	String (26)
apOrderService_sessionsRequestID	Set to the value of the requestID field returned in the sessions service response.	Order (R)	String (26)
apRefundService_reason	The reason for the refund.	Refund (O)	String (30)
apRefundService_refundRequestID	Set to the value of the requestID field returned in the sale or capture service response.	Refund (R)	String (26)
apRefundService_run	The refund service request name.	Refund (R)	String (26)
apSaleService_orderRequestID	Set to the value of the requestID field returned in the order service response.	Sale (R)	String (26)
apSaleService_run	The sale service request name.	Sale (R)	String (26)
apSessionsService_cancelURL	URL to which the customer is directed after canceling the payment.	Sessions (O)	String (255)
apSessionsService_paymentOptionID	Identifier for a PayPal credit transaction. Value: Credit	Sessions (R for PayPal credit transaction only)	String (255)
apSessionsService_run	The sessions service request name.	Sessions (R)	String (26)
apSessionsService_successURL	URL to which the customer is directed after completing the payment.	Sessions (O)	String (255)

Table 9 Request Fields (Continued)

Field	Description	Required (R) / Optional (O)	Data Type & Length
clientMetadataId	<p>Verifies that the payment is originating from a valid, user-approved application and device. Sending this field helps reduce fraud and declined transactions.</p> <p>Note The length is set for a hexadecimal representation of the GUID/UUID. This field accepts a 36-character string (with hyphens) or a 32-character string (without hyphens).</p> <p>Example 123e4567-e89b-12d3-a456-426655440000</p> <p>Example 123e4567e89b12d3a456426655440000</p>	Sale (R for reference transactions; otherwise O)	String (36)
invoiceHeader_merchantDescriptor	Merchant description on the customer's statement. When you include more than one consecutive space, extra spaces are removed.	Sale (O)	String (140)
item_#_productCode	<p>Type of product. This value is used to determine the product category: electronic, handling, physical, service, or shipping. The default value is <code>default</code>.</p> <p>For the authorization service, when you set this field to a value other than <code>default</code> or any of the values related to shipping and handling, the item_#_quantity, item_#_productName, and item_#_productSKU fields are required.</p>	<p>Sale (O; only for reference transactions)</p> <p>Sessions (O when creating an order; otherwise not used)</p>	String (255)
item_#_productDescription	<p>Description of the product.</p> <p>Note Do not include this field when creating a billing agreement.</p>	<p>Sale (O; only for reference transactions)</p> <p>Sessions (O when creating an order; otherwise not used)</p>	String (255)
item_#_productName	<p>Name of the product.</p> <p>For the authorization service and capture service, this field is required when item_#_productCode is not <code>default</code> or one of the values related to shipping and handling.</p> <p>Note Do not include this field when creating a billing agreement.</p>	<p>Sale (O; only for reference transactions)</p> <p>Sessions (O when creating an order; otherwise not used)</p>	String (255)

Table 9 Request Fields (Continued)

Field	Description	Required (R) / Optional (O)	Data Type & Length
item_#_productSKU	<p>Identification code for the product.</p> <p>For the authorization service and capture service, this field is required when item_#_productCode is not <code>default</code> or one of the values related to shipping and handling.</p> <p>Note Do not include this field when creating a billing agreement.</p>	<p>Sale (O; only for reference transactions)</p> <p>Sessions (O when creating an order; otherwise not used)</p>	String (255)
item_#_quantity	<p>The default is 1.</p> <p>For the authorization service and capture service, this field is required when item_#_productCode is not <code>default</code> or one of the values related to shipping and handling.</p> <p>Note Do not include this field when creating a billing agreement.</p>	<p>Sale (O; only for reference transactions)</p> <p>Sessions (O when creating an order; otherwise not used)</p>	Integer (10)
item_#_taxAmount	<p>Total tax to apply to the product. This value cannot be negative. The tax amount and the unit price must be in the same currency.</p> <p>The tax amount field is additive. The following example uses a two-exponent currency such as USD:</p> <ol style="list-style-type: none"> You include the following items in your request: <pre> item_0_unitPrice=10.00 item_0_quantity=1 item_0_taxAmount=0.80 item_1_unitPrice=20.00 item_1_quantity=1 item_1_taxAmount=1.60 </pre> The total amount authorized is 32.40, not 30.00 with 2.40 of tax included. 	<p>Sale (O; only for reference transactions)</p>	String (15)
item_#_unitPrice	<p>Per-item price of the product. This value cannot be negative. You can include a decimal point (.), but you cannot include any other special characters. CyberSource truncates the amount to the correct number of decimal places.</p> <p>Note Do not include this field when creating a billing agreement.</p>	<p>Sale (O; only for reference transactions)</p> <p>Sessions (O when creating an order; otherwise not used)</p>	String (15)

Table 9 Request Fields (Continued)

Field	Description	Required (R) / Optional (O)	Data Type & Length
item_#_unitTaxAmount	Per-item tax amount of the product. The amount value must be a non-negative number containing 2 decimal places and limited to 7 digits before the decimal point. Note Do not include this field when creating a billing agreement.	Sale (O; only for reference transactions) Sessions (O when creating an order; otherwise not used)	String (15)
merchantID	Your CyberSource merchant ID. Use the same merchant ID for evaluation, testing, and production.	Required for all services	String (30)
merchantReferenceCode	Merchant-generated transaction number or tracking number. CyberSource recommends that you send a unique value for each transaction so that you can perform meaningful searches for the transaction. For information about tracking orders and transactions, see Getting Started with CyberSource Advanced for the Simple Order API .	Required for all services	String (50)
note_toPayee	Note to the recipient of the funds in this transaction.	Sale (O)	String (255)
note_toPayer	Free-form text field.	Sale (O)	String (165)
purchaseTotals_currency	Currency for the transaction. Use the three-character ISO Standard Currency Codes . Important Decimal places are not supported for the Hungarian Forint (HUF) currency code. Note Do not include this field when creating a billing agreement.	Authorization (R) Capture (R) Order (R) Refund (R) Sale (R) Sessions (R when creating an order)	String (5)

Table 9 Request Fields (Continued)

Field	Description	Required (R) / Optional (O)	Data Type & Length
purchaseTotals_ grandTotalAmount	<p>Grand total for the transaction. This value cannot be negative. You can include a decimal point (.), but you cannot include any other special characters. CyberSource truncates the amount to the correct number of decimal places.</p> <p>The amount value must be a non-negative number containing 2 decimal places and limited to 7 digits before the decimal point.</p> <p>Note Do not include this field when creating a billing agreement.</p>	Authorization (R) Capture (R) Order (R) Refund (R) Sale (R) Sessions (R when creating an order)	String (10)
purchaseTotals_ handlingAmount	<p>Aggregate handling charges for the transaction.</p> <p>If this amount has changed since the initial sessions request, you must include the new value in the order request. You must also include all additional amount fields that apply to the order and ensure the total amount equals the purchaseTotals_grandTotalAmount value.</p> <p>Note The amount value must be a non-negative number containing 2 decimal places and limited to 7 digits before the decimal point.</p>	Order (O) Sale (O; only for a reference transaction) Sessions (R when creating an order)	String (15)
purchaseTotals_ shippingAmount	<p>Aggregate shipping charges for the transaction.</p> <p>If this amount has changed since the initial sessions request, you must include the new value in the order request. You must also include all additional amount fields that apply to the order and ensure the total amount equals the purchaseTotals_grandTotalAmount value.</p> <p>Note The amount value must be a non-negative number containing 2 decimal places and limited to 7 digits before the decimal point.</p>	Order (O) Sale (O; only for a reference transaction) Sessions (R when creating an order)	String (15)

Table 9 Request Fields (Continued)

Field	Description	Required (R) / Optional (O)	Data Type & Length
purchaseTotals_shippingDiscountAmount	<p>Shipping discount amount for the transaction.</p> <p>If this amount has changed since the initial sessions request, you must include the new value in the order request. You must also include all additional amount fields that apply to the order and ensure the total amount equals the purchaseTotals_grandTotalAmount value.</p> <p>Note The amount value must be a negative number containing 2 decimal places and limited to 7 digits before the decimal point.</p>	<p>Order (O)</p> <p>Sale (O)</p> <p>Sessions (O)</p>	String (15)
purchaseTotals_subtotalAmount	<p>Subtotal amount of all items.</p> <p>This amount (which is the value of all items in the cart, not including the additional amounts such as tax, shipping, etc.) cannot change after a sessions request. When there is a change to any of the additional amounts, this field should be resent in the order request. When the subtotal amount changes, you must initiate a new transaction starting with a sessions request.</p> <p>Note The amount value must be a non-negative number containing 2 decimal places and limited to 7 digits before the decimal point. This value can not be changed after a sessions request.</p>	<p>Order (O)</p> <p>Sale (O; only for a reference transaction)</p> <p>Sessions (R when creating an order)</p>	String (15)
purchaseTotals_taxAmount	<p>Total tax amount. When the purchaseTotals_taxAmount and ap_subtotalAmount fields are included in the request, do not include the tax amount as part of the subtotal amount calculation.</p> <p>Note The amount value must be a non-negative number containing 2 decimal places and limited to 7 digits before the decimal point.</p>	<p>Order (O)</p> <p>Sessions (R when creating an order)</p>	String (10)
shipTo_city	<p>City of the shipping address.</p> <p>Note This field is optional and valid only for a billing agreement (reference transaction).</p>	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (40)
shipTo_country	<p>Country of the shipping address. Use the two-character <i>ISO Standard Country Codes</i>.</p> <p>Note This field is optional and valid only for a billing agreement (reference transaction).</p>	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (2)

Table 9 Request Fields (Continued)

Field	Description	Required (R) / Optional (O)	Data Type & Length
shipTo_firstName	<p>First name of the recipient.</p> <p>Note This field is optional and valid only for a billing agreement (reference transaction).</p> <p>In-Store Collection When processing an in-store collection transaction in which the customer orders online and picks up the order in the store, avoid PayPal fraud declines by including S2S as the first 3 characters. A blank space must also be included between S2S and the rest of the first name information.</p> <p>Example S2S John</p>	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (60)
shipTo_immutable	<p>Indicates whether customers are permitted to edit the shipping address in their PayPal account. Possible values:</p> <ul style="list-style-type: none"> ■ <code>true</code>: Customer cannot edit the shipping address. ■ <code>false</code> (default): Customer can edit the shipping address. 	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (5)
shipTo_lastName	<p>Last name of the recipient.</p> <p>Note This field is optional and valid only for a billing agreement (reference transaction).</p>	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (60)
shipTo_notApplicable	<p>Indicates whether the shipping address is displayed to the customer in their PayPal account. Possible values:</p> <ul style="list-style-type: none"> ■ <code>true</code>: Shipping address is not displayed. ■ <code>false</code> (default): Shipping address is displayed. <p>For example, for digital downloads and services in which a shipping address is not required, set the value to <code>true</code>.</p>	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (5)
shipTo_phoneNumber	<p>Phone number for the shipping address.</p> <p>Note This field is optional and valid only for a billing agreement (reference transaction).</p>	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (20)

Table 9 Request Fields (Continued)

Field	Description	Required (R) / Optional (O)	Data Type & Length
shipTo_postalCode	<p>Postal code for the shipping address. The postal code must consist of 5 to 9 digits.</p> <p>When the shipping country is the U.S., the 9-digit postal code must follow this format: [5 digits][dash][4 digits]</p> <p>Example 12345-6789</p> <p>When the shipping country is Canada, the 6-digit postal code must follow this format: [alpha][numeric][alpha][space][numeric][alpha][numeric]</p> <p>Example A1B 2C3</p> <p>Note This field is optional and valid only for a billing agreement (reference transaction).</p>	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (20)
shipTo_state	<p>State or province of the shipping address. Use the State, Province, and Territory Codes for the United States and Canada.</p> <p>Note This field is optional and valid only for a billing agreement (reference transaction).</p>	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (40)
shipTo_street1	<p>First line of the shipping address.</p> <p>Note This field is optional and valid only for a billing agreement (reference transaction).</p>	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (100)
shipTo_street2	<p>Second line of the shipping address.</p> <p>Note This field is optional and valid only for a billing agreement (reference transaction).</p>	<p>Sale (O; only for a reference transaction)</p> <p>Sessions (O)</p>	String (100)

Reply Fields

Table 10 Reply Fields

Field	Description	Returned By	Data Type & Length
apAuthReply_amount	The authorized amount.	Authorization	String (15)
apAuthReply_dateTime	Time of authorization. Format: YY YY-MM-DDThh:mm:ssZ Example 2019-08-11T22:47:57Z equals August 11, 2019, at 22:47:57 (10:47:57 p.m.). The T separates the date and the time. The Z indicates UTC.	Authorization	String (18)
apAuthReply_paymentStatus	Authorization status from the processor. Possible values: <ul style="list-style-type: none"> ▪ Pending ▪ Authorized ▪ Expired ▪ Failed 	Authorization	String (15)
apAuthReply_processorResponse	The processor code that describes why the transaction state is pending or reversed. Note The returned value can also include the processor response values. See Appendix D, "Reason Codes," on page 79 .	Authorization	String (60)
apAuthReply_processorTransactionID	Identifier of the order transaction. Read-only.	Authorization	String (50)
apAuthReply_reasonCode	Numeric value corresponding to the result of the authorization service request.	Authorization	Integer (5)
apAuthReply_reconciliationID	Reference number that you use to reconcile your CyberSource reports with your business reports.	Authorization	String (60)
apAuthReply_responseCode	Response code from the processor.	Authorization	Integer (5)
apAuthReversalReply_amount	The authorized amount that was reversed.	Auth Reversal	String (15)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apAuthReversalReply_dateTime	Date and time of authorization reversal. Format: YYYY-MM-DDThh:mm:ssZ Example 2019-08-11T22:47:57Z equals August 11, 2019, at 22:47:57 (10:47:57 p.m.). The T separates the date and the time. The Z indicates UTC.	Auth Reversal	String (18)
apAuthReversalReply_paymentStatus	The payment status returned from the payment processor. Possible values: <ul style="list-style-type: none"> ■ Auth_reversed ■ Failed 	Auth Reversal	String (15)
apAuthReversalReply_processorResponse	The processor code that describes why the transaction state is pending or reversed. Note The returned value can also include the processor response values. See Appendix D, "Reason Codes," on page 79 .	Auth Reversal	String (60)
apAuthReversalReply_processorTransactionID	Identifier of the order transaction. Read-only.	Auth Reversal	String (50)
apAuthReversalReply_reasonCode	Numeric value corresponding to the result of the auth reversal service request.	Auth Reversal	Integer (5)
apAuthReversalReply_reconciliationID	Reference number that you use to reconcile your CyberSource reports with your business reports.	Auth Reversal	String (60)
apAuthReversalReply_responseCode	Response code from the processor.	Auth Reversal	Integer (5)
apBillingAgreementReply_amount	Total amount charged from the payer to the payee. Maximum of 10 characters with support for 2 decimal places.	Billing Agreement	String(15)
apBillingAgreementReply_dateTime	Time of billing agreement. Format: YYYY-MM-DDThh:mm:ssZ Example 2019-08-11T22:47:57Z equals August 11, 2019, at 22:47:57 (10:47:57 p.m.). The T separates the date and the time. The Z indicates UTC.	Billing Agreement	String (18)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apBillingAgreementReply_processorResponse	<p>Response code indicating that creating the agreement failed.</p> <p>Note The returned value can also include the processor response values.</p> <p>See Appendix D, "Reason Codes," on page 79.</p>	Billing Agreement	String(60)
apBillingAgreementReply_reasonCode	Numeric value corresponding to the result of the billing agreement service request.	Billing Agreement	Integer (5)
apBillingAgreementReply_status	<p>State of the order transaction.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ■ Active ■ Inactive ■ Failed 	Billing Agreement	String(60)
apCancelReply_dateTime	<p>Time of cancel.</p> <p>Format: YYYY-MM-DDThh:mm:ssZ</p> <p>Example 2019-08-11T22:47:57 equals August 11, 2019, at 22:47:57 (10:47:57 p.m.). The T separates the date and the time. The Z indicates UTC.</p>	Cancel	String (18)
apCancelReply_processorResponse	<p>The reason when the transaction status is Pending or Reversed.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ■ PAYER_SHIPPING_UNCONFIRMED ■ MULTI_CURRENCY ■ RISK_REVIEW ■ REGULATORY_REVIEW ■ VERIFICATION_REQUIRED ■ ORDER ■ OTHER <p>Note The returned value can also include the processor response values.</p> <p>See Appendix D, "Reason Codes," on page 79.</p>	Cancel	String (60)
apCancelReply_processorTransactionID	<p>Identifier of the order transaction.</p> <p>Read-only.</p>	Cancel	String (50)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apCancelReply_reasonCode	Numeric value corresponding to the result of the cancel service request. See Appendix D, "Reason Codes," on page 79 .	Cancel	Integer (5)
apCancelReply_reconciliationID	Reference number that you use to reconcile your CyberSource reports with your reports.	Cancel	String (60)
apCancelReply_status	CYBS mapped values. Possible values: <ul style="list-style-type: none"> ■ Canceled ■ Failed 	Cancel	String (60)
apCaptureReply_reasonCode	Numeric value corresponding to the result of the capture service request.	Capture	Integer (5)
apCaptureReply_dateTime	Time of capture. Format: YYYY-MM-DDThh:mm:ssZ Example 2019-08-11T22:47:57Z equals August 11, 2019, at 22:47:57 (10:47:57 p.m.). The T separates the date and the time. The Z indicates UTC.	Capture	String (18)
apCaptureReply_paymentStatus	Description of the payment status. Possible values: <ul style="list-style-type: none"> ■ Pending ■ Settled ■ Failed 	Capture	String (15)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apCaptureReply_processorResponse	<p>The processor code that describes why the transaction state is pending or reversed.</p> <p>Possible values when the payment status is <i>pending</i>:</p> <ul style="list-style-type: none"> ■ UNCONFIRMED_SHIPPING_ADDRESS ■ INTERNATIONAL_WITHDRAWAL ■ RECEIVING_PREFERENCE_MANDATES_MANUAL_ACTION ■ PAYMENT_REVIEW ■ REGULATORY_REVIEW ■ VERIFICATION_REQUIRED ■ TRANSACTION_APPROVED_AWAITING_FUNDING <p>Note The returned value can also include the processor response values. See Appendix D, "Reason Codes," on page 79.</p>	Capture	String (60)
apCaptureReply_processorTransactionFee	Amount up to N digit after the decimal separator as defined in ISO 4217 for the appropriate currency code.	Capture	String (15)
apCaptureReply_processorTransactionID	<p>Identifier of the order transaction.</p> <p>Read-only.</p>	Capture	String (50)
apCaptureReply_reconciliationID	Reference number that you use to reconcile your CyberSource reports with your business reports.	Capture	String (60)
apCheckStatusReply_reasonCode	Numeric value corresponding to the result of the check status service request.	Check Status	Integer (5)
apCheckStatusReply_dateTime	<p>Time of check status.</p> <p>Format: YYYY-MM-DDThh:mm:ssZ</p> <p>Example 2019-08-11T22:47:57Z equals August 11, 2019, at 22:47:57 (10:47:57 p.m.). The T separates the date and the time. The Z indicates UTC.</p>	Check Status	String (18)
apCheckStatusReply_paymentStatus	<p>Description of the payment status.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ■ Created ■ Failed 	Check Status	String (15)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apCheckStatusReply_processorResponse	The value returned by the processor indicating the reason the request failed. See Appendix D, "Reason Codes," on page 79.	Check Status	String (60)
apCheckStatusReply_processorTransactionID	Identifier of the order transaction. Read only.	Check Status	String (50)
apCheckStatusReply_reconciliationID	Reference number that you use to reconcile your CyberSource reports with your business reports.	Check Status	String (60)
apOrderReply_amount	Total amount charged from the customer to the merchant. In the case of a refund, this is the amount refunded to the customer from the merchant. Ten-character maximum with 0 to 2 decimal places.	Order	String (15)
apOrderReply_dateTime	Time of order. Format: YYYY-MM-DDThh:mm:ssZ Example 2019-08-11T22:47:57Z equals August 11, 2019, at 22:47:57 (10:47:57 p.m.). The T separates the date and the time. The Z indicates UTC.	Order	String (18)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apOrderReply_ processorResponse	<p>The reason for the payment failure.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ■ UNABLE_TO_COMPLETE_TRANSACTION ■ INVALID_PAYMENT_METHOD ■ PAYER_CANNOT_PAY ■ CANNOT_PAY_THIS_PAYEE ■ REDIRECT_REQUIRED ■ PAYEE_FILTER_RESTRICTIONS ■ PAYER_SHIPPING_UNCONFIRMED ■ MULTI_CURRENCY ■ RISK_REVIEW ■ REGULATORY_REVIEW ■ VERIFICATION_REQUIRED ■ ORDER ■ OTHER <p>Note The returned value can also include the processor response values. See Appendix D, "Reason Codes," on page 79.</p>	Order	String (60)
apOrderReply_ reasonCode	<p>Numeric value corresponding to the result of the order service request.</p> <p>See Appendix D, "Reason Codes," on page 79.</p>	Order	Integer (5)
apOrderReply_ reconciliationID	Invoice number to track the payment.	Order	String (127)
apOrderReply_ status	<p>State of the order transaction.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ■ Created ■ Cancelled ■ Failed 	Order	String (60)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apRefundReply_dateTime	Time of refund. Format: YYYY-MM-DDThh:mm:ssZ Example 2019-08-11T22:47:57Z equals August 11, 2019, at 22:47:57 (10:47:57 p.m.). The T separates the date and the time. The Z indicates UTC.	Refund	String (18)
apRefundReply_paymentStatus	Refund status from the processor. Possible values: <ul style="list-style-type: none">■ Pending■ Refunded■ Failed	Refund	String (15)
apRefundReply_processorResponse	The processor code that describes why the transaction state is pending or reversed. Note The returned value can also include the processor response values. See Appendix D, "Reason Codes," on page 79.	Refund	String (60)
apRefundReply_processorTransactionID	Identifier of the order transaction. Read-only.	Refund	String (50)
apRefundReply_reasonCode	Numeric value corresponding to the result of the refund service request. See Appendix D, "Reason Codes," on page 79.	Refund	Integer (5)
apRefundReply_reconciliationID	Reference number that you use to reconcile your CyberSource reports with your business reports.	Refund	String (60)
apReply_billingAgreementID	The Billing Agreement ID returned by processor (PayPal).	Billing Agreement	String(50)
apReply_fundingSource	Payment mode for the transaction. Possible values: <ul style="list-style-type: none">■ INSTANT_TRANSFER■ MANUAL_BANK_TRANSFER■ DELAYED_TRANSFER■ ECHECK	Authorization Check Status Order Sale Sessions	String (30)
apReply_orderID	Identifier for the purchase.	Order	String (20)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apReply_payerID	The ID of the customer, passed in the return_url field by PayPal after customer approval.	Billing Agreement Check Status Order	String(30)
apReply_processorFraudDecision	Type of filter. Possible values: <ul style="list-style-type: none"> ■ ACCEPT ■ PENDING ■ DENY ■ REPORT 	Authorization Order Sale	String (60)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apReply_ processorFraudDecision Reason	Filter Identifier. Possible values: <ul style="list-style-type: none"> ■ AVS_NO_MATCH ■ AVS_PARTIAL_MATCH ■ AVS_UNAVAILABLE_OR_UNSUPPORTED ■ CARD_SECURITY_CODE_MISMATCH ■ MAXIMUM_TRANSACTION_AMOUNT ■ UNCONFIRMED_ADDRESS ■ COUNTRY_MONITOR ■ LARGE_ORDER_NUMBER ■ BILLING_OR_SHIPPING_ADDRESS_MISMATCH ■ RISKY_ZIP_CODE ■ SUSPECTED_FREIGHT_FORWARDER_CHECK ■ TOTAL_PURCHASE_PRICE_MINIMUM ■ IP_ADDRESS_VELOCITY ■ RISKY_EMAIL_ADDRESS_DOMAIN_CHECK ■ RISKY_BANK_IDENTIFICATION_NUMBER_CHECK ■ RISKY_IP_ADDRESS_RANGE ■ PAYPAL_FRAUD_MODEL 	Authorization Order Sale	String (60)
apReply_sellerProtection_ eligibility	The level of seller protection in force for the transaction. Possible values: <ul style="list-style-type: none"> ■ ELIGIBLE ■ PARTIALLY_ELIGIBLE ■ INELIGIBLE 	Authorization Check Status Order Sale	String (60)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apReply_sellerProtection_type	<p>The kind of seller protection in force for the transaction. This field is returned only when the protection_eligibility property is set to ELIGIBLE or PARTIALLY_ELIGIBLE.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ■ ITEM_NOT_RECEIVED_ELIGIBLE—Sellers are protected against claims for items not received. ■ UNAUTHORIZED_PAYMENT_ELIGIBLE—Sellers are protected against claims for unauthorized payments. <p>One or both values can be returned.</p>	<p>Authorization</p> <p>Check Status</p> <p>Order</p> <p>Sale</p>	String (60)
apSaleReply_amount	Total amount charged from the payer to the payee. Maximum of 10 characters with support for 2 decimal places.	Sale	String (15)
apSaleReply_dateTime	<p>Time of sale.</p> <p>Format: YYYY-MM-DDThh:mm:ssZ</p> <p>Example 2019-08-11T22:47:57Z equals August 11, 2019, at 22:47:57 (10:47:57 p.m.). The T separates the date and the time. The Z indicates UTC.</p>	Sale	String (18)
apSaleReply_paymentStatus	<p>Description of the payment status.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ■ Pending ■ Settled ■ Failed 	Sale	String (15)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apSaleReply_ processorResponse	<p>The processor code that describes why the transaction state is pending or reversed.</p> <p>Possible values:</p> <ul style="list-style-type: none"> ■ CHARGEBACK ■ GUARANTEE ■ BUYER_COMPLAINT ■ ECHECK ■ INTERNATIONAL_WITHDRAWAL ■ RECEIVING_PREFERENCE_MANDATES_MANUAL_ACTION ■ PAYMENT_REVIEW ■ REGULATORY_REVIEW ■ UNILATERAL ■ UNCONFIRMED_SHIPPING_ADDRESS ■ VERIFICATION_REQUIRED ■ TRANSACTION_APPROVED_AWAITING_FUNDING <p>Note The returned value can also include the processor response values. See Appendix D, "Reason Codes," on page 79.</p>	Sale	String (60)
apSaleReply_ processorTransactionFee	Amount up to N digit after the decimal separator as defined in ISO 4217 for the appropriate currency code.	Sale	String (15)
apSaleReply_ processorTransactionID	Identifier of the order transaction. Read-only.	Sale	String (50)
apSaleReply_reasonCode	Numeric value corresponding to the result of the sale service request. See Appendix D, "Reason Codes," on page 79.	Sale	Integer (5)
apSaleReply_ reconciliationID	Reference number that you use to reconcile your CyberSource reports with your reports.	Sale	String (60)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
apSessionsReply_amount	Total amount charged from the payer to the payee. In the case of a refund, this is the refunded amount to the original payer from the payee. Maximum of 10 characters with support for 2 decimal places.	Sessions	String (15)
apSessionsReply_dateTime	Time of session. Format: YYYY-MM-DDThh:mm:ssZ Example 2019-08-11T22:47:57Z equals August 11, 2019, at 22:47:57 (10:47:57 p.m.). The T separates the date and the time. The Z indicates UTC.	Sessions	String (18)
apSessionsReply_merchantURL	Redirect URL to PayPal web site.	Sessions	String (2048)
apSessionsReply_paymentStatus	Status of the payment request. Possible values: <ul style="list-style-type: none"> ■ Created ■ Failed 	Sessions	String (15)
apSessionsReply_processorTransactionID	ID of the created payment. Value assigned by PayPal.	Sessions	String (50)
apSessionsReply_reasonCode	Numeric value corresponding to the result of the sessions service request.	Sessions	Integer (5)
apSessionsReply_reconciliationID	Invoice number to track this payment. Maximum length: 127.	Sessions	String(127)
apSessionsReply_responseCode	The transaction response code received from the processor.	Sessions	String (5)
apSessionsReply_status	Status of the sessions request. Possible values: <ul style="list-style-type: none"> ■ Created ■ Failed 	Sessions	String (15)
billTo_city	City of the billing address. Note To get the details of a customer's billing address from PayPal using the check status service, you must have this feature enabled on your PayPal account. Contact your PayPal account manager.	Billing Agreement Check Status Order	String (50)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
billTo_country	Country of the billing address. Use the two-character <i>ISO Standard Country Codes</i> . Note To get the details of a customer's billing address from PayPal using the check status service, you must have this feature enabled on your PayPal account. Contact your PayPal account manager.	Billing Agreement Check Status Order	String (2)
billTo_email	Customer's email address. Note To get the details of a customer's billing address from PayPal using the check status service, you must have this feature enabled on your PayPal account. Contact your PayPal account manager.	Billing Agreement Check Status Order	String (256)
billTo_firstName	First name of customer. Note To get the details of a customer's billing address from PayPal using the check status service, you must have this feature enabled on your PayPal account. Contact your PayPal account manager.	Billing Agreement Check Status Order	String (60)
billTo_lastName	Last name of customer. Note To get the details of a customer's billing address from PayPal using the check status service, you must have this feature enabled on your PayPal account. Contact your PayPal account manager.	Billing Agreement Check Status Order	String (60)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
billTo_postalCode	<p>Postal code for the billing address. The postal code must consist of 5 to 9 digits.</p> <p>When the billing country is the U.S., the 9-digit postal code must follow this format: [5 digits][dash][4 digits]</p> <p>Example 12345-6789</p> <p>When the billing country is Canada, the 6-digit postal code must follow this format: [alpha][numeric][alpha][space][numeric][alpha][numeric]</p> <p>Example A1B 2C3</p> <p>Note To get the details of a customer's billing address from PayPal using the check status service, you must have this feature enabled on your PayPal account. Contact your PayPal account manager.</p>	<p>Billing Agreement</p> <p>Check Status</p> <p>Order</p>	String (10)
billTo_state	<p>State or province of the billing address. Use the State, Province, and Territory Codes for the United States and Canada.</p> <p>Note To get the details of a customer's billing address from PayPal using the check status service, you must have this feature enabled on your PayPal account. Contact your PayPal account manager.</p>	<p>Billing Agreement</p> <p>Check Status</p> <p>Order</p>	String (2)
billTo_street1	<p>First line of the billing street address.</p> <p>Note To get the details of a customer's billing address from PayPal using the check status service, you must have this feature enabled on your PayPal account. Contact your PayPal account manager.</p>	<p>Billing Agreement</p> <p>Check Status</p> <p>Order</p>	String (60)
billTo_street2	<p>Second line of the billing street address.</p> <p>Note To get the details of a customer's billing address from PayPal using the check status service, you must have this feature enabled on your PayPal account. Contact your PayPal account manager.</p>	<p>Billing Agreement</p> <p>Check Status</p> <p>Order</p>	String (60)
customerVerificationStatus	<p>Possible values are VERIFIED or UNVERIFIED, depending on whether the buyer has verified their identity. Used in PayPal transactions.</p>	<p>Check Status</p>	String (60)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
decision	Summarizes the result of the overall request. For information about handling replies, see Getting Started with CyberSource Advanced for the Simple Order API . Possible values: <ul style="list-style-type: none"> ■ ACCEPT ■ ERROR ■ REJECT ■ REVIEW 	All services	String (6)
invalidField_0 through invalidField_N	Fields in the request that have invalid data. For information about missing or invalid fields, see Getting Started with CyberSource Advanced for the Simple Order API .	Authorization Auth Reversal Billing Agreement Cancel Capture Check Status Order Refund Sale Sessions	String (100)
reasonCode	Numeric value corresponding to the result of the overall request. See Appendix D, "Reason Codes," on page 79 .	All services	Integer (5)
requestID	Identifier for the request generated by the client.	All services	String (26)
shipTo_city	City in the shipping address.	Billing Agreement Check Status Order	String (40)
shipTo_country	Country in the shipping address. This is a two-character ISO Standard Country Codes .	Billing Agreement Check Status Order	String (2)
shipTo_firstName	First name of recipient.	Billing Agreement Check Status Order	String (60)

Table 10 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
shipTo_lastName	Last name of the recipient.	Billing Agreement Check Status Order	String (60)
shipTo_phoneNumber	Phone number of the recipient.	Billing Agreement Check Status Order	String (20)
shipTo_postalCode	Postal code of shipping address. Consists of 5 to 9 digits.	Billing Agreement Check Status Order	String (20)
shipTo_state	State or province in the shipping address. This is a State, Province, and Territory Codes for the United States and Canada .	Billing Agreement Check Status Order	String (40)
shipTo_street1	First line of the shipping address.	Billing Agreement Check Status Order	String (100)
shipTo_street2	Second line of the shipping address.	Billing Agreement Check Status Order	String (100)

Service Examples

Sessions Service

Example 2 Sessions Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <invoiceHeader>
    <merchantDescriptor>MerchantDesc</merchantDescriptor>
  </invoiceHeader>
  <shipTo>
    <firstName>Jane</firstName>
    <lastName>Smith</lastName>
    <street1>123 Main Street</street1>
    <street2>Mailstop A</street2>
    <city>Small Town</city>
    <state>CA</state>
    <postalCode>98765</postalCode>
    <country>US</country>
    <phoneNumber>9876543210</phoneNumber>
    <shippingMethod>FedEx</shippingMethod>
  </shipTo>
  <item id="0">
    <unitPrice>45.00</unitPrice>
    <quantity>2</quantity>
    <productCode>default</productCode>
    <productName>TestProduct</productName>
    <productSKU>TestSKU</productSKU>
    <productDescription>TestDesc</productDescription>
    <unitTaxAmount>1.00</unitTaxAmount>
  </item>
  <purchaseTotals>
    <currency>USD</currency>
    <taxAmount>2.00</taxAmount>
    <grandTotalAmount>95.00</grandTotalAmount>
    <subtotalAmount>90.00</subtotalAmount>
    <shippingAmount>5.00</shippingAmount>
    <handlingAmount>3.00</handlingAmount>
    <shippingDiscountAmount>5.00</shippingDiscountAmount>
  </purchaseTotals>
```

```

<apPaymentType>PPL</apPaymentType>
<apSessionsService run="true">
  <cancelURL>http://www.example.com</cancelURL>
  <successURL>http://www.example.com</successURL>
  <reconciliationID>12345</reconciliationID>
</apSessionsService>
</requestMessage>

```

Example 3 Sessions Reply

```

<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>4980396424596224104012</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:purchaseTotals>
    <c:currency>USD</c:currency>
  </c:purchaseTotals>
  <c:apSessionsReply>
    <c:reasonCode>100</c:reasonCode>
    <c:merchantURL>https://www.sandbox.paypal.com/cgi-bin/webscr?cmd=_express-checkout&token=EC-1YN71306R29977435</c:merchantURL>
    <c:processorTransactionID>PAY-7PN460756LFFEKWY</c:processorTransactionID>
    <c:amount>95.00</c:amount>
    <c:reconciliationID>40HXQXLD0000J3XKJCQV</c:reconciliationID>
    <c:status>CREATED</c:status>
  </c:apSessionsReply>
</c:replyMessage>

```

Order Service

Example 4 Order Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <item id="0">
    <unitPrice>45.00</unitPrice>
    <quantity>2</quantity>
    <productName>TestProduct</productName>
    <productCode>default</productCode>
    <productSKU>TestSKU</productSKU>
    <productDescription>TestDesc</productDescription>
    <unitTaxAmount>1.00</unitTaxAmount>
  </item>
  <purchaseTotals>
    <currency>USD</currency>
    <taxAmount>2.00</taxAmount>
    <grandTotalAmount>95.00</grandTotalAmount>
    <subtotalAmount>90.00</subtotalAmount>
    <shippingAmount>5.00</shippingAmount>
    <handlingAmount>3.00</handlingAmount>
    <shippingDiscountAmount>5.00</shippingDiscountAmount>
  </purchaseTotals>
  <apPaymentType>PPL</apPaymentType>
  <ap>
    <payerID>7FUDAXBNR5KSC</payerID>
  </ap>
  <apOrderService run="true">
    <sessionsRequestID>4980396424596224104012</sessionsRequestID>
  </apOrderService>
</requestMessage>
```

Example 5 Order Reply

```

<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>4980405898776291704008</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:purchaseTotals>
    <c:currency>USD</c:currency>
  </c:purchaseTotals>
  <c:apReply>
    <c:orderID>O-1CM486978D015523P</c:orderID>
    <c:payerID>7FUDAXBNR5KSC</c:payerID>
  </c:apReply>
  <c:shipTo>
    <c:firstName>Jane</c:firstName>
    <c:lastName>Smith</c:lastName>
    <c:street1>123 Main Street</c:street1>
    <c:street2>Mailstop A</c:street2>
    <c:city>Small Town</c:city>
    <c:state>CA</c:state>
    <c:postalCode>98765</c:postalCode>
    <c:country>US</c:country>
  </c:shipTo>
  <c:billTo>
    <c:firstName>Jane</c:firstName>
    <c:lastName>Smith</c:lastName>
    <c:country>US</c:country>
    <c:email>jsmith@example.com</c:email>
  </c:billTo>
  <c:apOrderReply>
    <c:reasonCode>100</c:reasonCode>
    <c:amount>100.00</c:amount>
    <c:reconciliationID>40HXQXLD0000J3XKJCQV</c:reconciliationID>
    <c:status>CREATED</c:status>
    <c:processorResponse>ORDER</c:processorResponse>
  </c:apOrderReply>
</c:replyMessage>

```

Authorization Service

Example 6 Authorization Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <purchaseTotals>
    <currency>USD</currency>
    <grandTotalAmount>95.00</grandTotalAmount>
  </purchaseTotals>
  <apPaymentType>PPL</apPaymentType>
  <apAuthService run="true">
    <orderRequestID>4980405898776291704008</orderRequestID>
  </apAuthService>
</requestMessage>
```

Example 7 Authorization Reply

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>4896155375886014701200</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:purchaseTotals>
    <c:currency>USD</c:currency>
  </c:purchaseTotals>
  <c:apAuthReply>
    <c:reasonCode>100</c:reasonCode>
    <c:transactionID>4KP97418B91840630</c:transactionID>
    <c:amount>95.00</c:amount>
    <c:paymentStatus>AUTHORIZED</c:paymentStatus>
    <c:reconciliationID>ZZEWULIG00000000000000000000000000000000IZWESA00</c:reconciliationID>
    <c:processorTransactionID>4KP97418B91840630</c:processorTransactionID>
  </c:apAuthReply>
</c:replyMessage>
```

Capture Service

Example 10 Capture Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <purchaseTotals>
    <currency>USD</currency>
    <grandTotalAmount>95.00</grandTotalAmount>
  </purchaseTotals>
  <apPaymentType>PPL</apPaymentType>
  <apCaptureService run="true">
    <authRequestID>4896155375886014701200</authRequestID>
  </apCaptureService>
</requestMessage>
```

Example 11 Capture Reply

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>6886190757355555601201</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:purchaseTotals>
    <c:currency>USD</c:currency>
  </c:purchaseTotals>
  <c:apCaptureReply>
    <c:reasonCode>100</c:reasonCode>
    <c:processorTransactionID>30M24208823051238</c:processorTransactionID>
    <c:processorResponse>None</c:processorResponse>
    <c:amount>95.00</c:amount>
    <c:reconciliationID>ZZEWULIG00000000000000000000IZWESA0B</c:reconciliationID>
    <c:paymentStatus>SETTLED</c:paymentStatus>
    <c:responseCode>None</c:responseCode>
  </c:apCaptureReply>
</c:replyMessage>
```

Sale Service

Example 12 Sale Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <purchaseTotals>
    <currency>USD</currency>
    <grandTotalAmount>95.00</grandTotalAmount>
  </purchaseTotals>
  <apPaymentType>PPL</apPaymentType>
  <apSaleService run="true">
    <orderRequestID>6886190757355555601201</orderRequestID>
  </apSaleService>
</requestMessage>
```

Example 13 Sale Reply

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>1124439821106016799952</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:purchaseTotals>
    <c:currency>USD</c:currency>
  </c:purchaseTotals>
  <c:apSaleReply>
    <c:reasonCode>100</c:reasonCode>
    <c:paymentStatus>SETTLED</c:paymentStatus>
    <c:responseCode>None</c:responseCode>
    <c:processorTransactionID>8HC63549VB1392407</c:processorTransactionID>
    <c:reconciliationID>ZZEWULIG00000000000000000000IZWESAOF</c:reconciliationID>
    <c:processorTransactionFee>3.20</c:processorTransactionFee>
    <c:amount>95.00</c:amount>
    <c:processorResponse>None</c:processorResponse>
  </c:apSaleReply>
</c:replyMessage>
```

Refund Service

Example 14 Refund Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <purchaseTotals>
    <currency>USD</currency>
    <grandTotalAmount>95.00</grandTotalAmount>
  </purchaseTotals>
  <apPaymentType>PPL</apPaymentType>
  <apRefundService run="true">
    <refundRequestID>1124439821106016799952</refundRequestID>
  </apRefundService>
</requestMessage>
```

Example 15 Refund Reply

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>48961921504567098712009999</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:purchaseTotals>
    <c:currency>USD</c:currency>
  </c:purchaseTotals>
  <c:apRefundReply>
    <c:reasonCode>100</c:reasonCode>
    <c:transactionID>8FU4435670486773H</c:transactionID>
    <c:amount>95.00</c:amount>
    <c:reconciliationID>ZZEWULIG00000000000000000000IZWESAOC</c:reconciliationID>
    <c:processorTransactionID>8FU4435670486773H</c:processorTransactionID>
    <c:paymentStatus>REFUNDED</c:paymentStatus>
  </c:apRefundReply>
</c:replyMessage>
```

Cancel Service

Example 16 Cancel Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <apPaymentType>PPL</apPaymentType>
  <apCancelService run="true">
    <orderRequestID>1124439821106016799952</orderRequestID>
  </apCancelService>
</requestMessage>
```

Example 17 Cancel Reply

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>3333330307786014000000</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:purchaseTotals>
    <c:currency>USD</c:currency>
  </c:purchaseTotals>
  <c:apCancelReply>
    <c:reasonCode>100</c:reasonCode>
    <c:processorTransactionID>O-7W241641VD265833L</c:processorTransactionID>
    <c:status>CANCELLED</c:status>
  </c:apCancelReply>
</c:replyMessage>
```

Billing Agreement Service

Example 18 Billing Agreement Service Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.133">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <apPaymentType>PPL</apPaymentType>
  <apBillingAgreementService run="true">
    <sessionsRequestID>4980396424596224104012</sessionsRequestID>
  </apBillingAgreementService>
</requestMessage>
```

Example 19 Billing Agreement Service Reply

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.133">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>4955835535216022201540</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:apReply>
    <c:billingAgreementID>SOK-3FJ6025</c:billingAgreementID>
  </c:apReply>
  <c:shipTo>
    <c:firstName>Jane</c:firstName>
    <c:lastName>Smith</c:lastName>
    <c:street1>123 Main St</c:street1>
    <c:street2>Mailstop A</c:street2>
    <c:city>Small Town</c:city>
    <c:state>CA</c:state>
    <c:postalCode>98765</c:postalCode>
    <c:country>US</c:country>
    <c:phoneNumber>9876543210</c:phoneNumber>
  </c:shipTo>
  <c:billTo>
    <c:firstName>Jane</c:firstName>
    <c:lastName>Smith</c:lastName>
    <c:street1>123 Main Street</c:street1>
    <c:street2>Mailstop A</c:street2>
    <c:city>Small Town</c:city>
    <c:state>CA</c:state>
    <c:postalCode>98765</c:postalCode>
    <c:country>US</c:country>
    <c:phoneNumber>9876543210</c:phoneNumber>
    <c:email>jsmith@example.com</c:email>
  </c:billTo>
  <c:apBillingAgreementReply>
    <c:reasonCode>100</c:reasonCode>
    <c:status>ACTIVE</c:status>
  </c:apBillingAgreementReply>
</c:replyMessage>
```

Check Status Examples

Sessions Service

Example 20 Check Status Request: Sessions Service

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <apPaymentType>PPL</apPaymentType>
  <apCheckStatusService run="true">
    <sessionsRequestID>4980396424596224104012</sessionsRequestID>
  </apCheckStatusService>
</requestMessage>
```

Example 21 Check Status Reply: Sessions Service

```

<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>4980396424596224104012</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:apCheckStatusReply>
    <c:reasonCode>100</c:reasonCode>
    <c:reconciliationID>ZZEWULIG00000000000000000000IZWESAOG</c:reconciliationID>
    <c:paymentStatus>CREATED</c:paymentStatus>
    <c:processorTransactionID>PAY-78U923845790YLDE5KZY</c:processorTransactionID>
  </c:apCheckStatusReply>
  <c:apReply>
    <c:payerID>7FUDAXBNR5KSC</c:payerID>
  </c:apReply>
  <c:shipTo>
    <c:firstName>Jane</c:firstName>
    <c:lastName>Smith</c:lastName>
    <c:street1>123 Main Street</c:street1>
    <c:street2>Mailstop A</c:street2>
    <c:city>Small Town</c:city>
    <c:state>CA</c:state>
    <c:country>US</c:country>
    <c:phoneNumber>9876543210</c:phoneNumber>
    <c:postalCode>98765</c:postalCode>
  </c:shipTo>
  <c:billTo>
    <c:firstName>Jane</c:firstName>
    <c:lastName>Smith</c:lastName>
    <c:street1>123 Main St</c:street1>
    <c:city>Small Town</c:city>
    <c:state>CA</c:state>
    <c:country>US</c:country>
    <c:postalCode>98765</c:postalCode>
    <c:email>jsmith@example.com</c:email>
  </c:billTo>
  <c:customerVerificationStatus>UNVERIFIED</c:customerVerificationStatus>
</c:replyMessage>

```

Order Service

Example 22 Check Status Request: Order Service

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <apPaymentType>PPL</apPaymentType>
  <apCheckStatusService run="true">
    <checkStatusRequestID>4980405898776291704008</checkStatusRequestID>
  </apCheckStatusService>
</requestMessage>
```

Example 23 Check Status Reply: Order Service

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>4896415760916002201200</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:apCheckStatusReply>
    <c:reasonCode>100</c:reasonCode>
    <c:paymentStatus>CREATED</c:paymentStatus>
    <c:processorTransactionID>O-5TD10988AD299315S</c:processorTransactionID>
  </c:apCheckStatusReply>
</c:replyMessage>
```

Authorization Service

Example 24 Check Status Request: Authorization Service

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <apPaymentType>PPL</apPaymentType>
  <apCheckStatusService run="true">
    <checkStatusRequestID>4896155375886014701200</checkStatusRequestID>
  </apCheckStatusService>
</requestMessage>
```

Example 25 Check Status Reply: Authorization Service

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>4896452294406002901200</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:apCheckStatusReply>
    <c:reasonCode>100</c:reasonCode>
    <c:paymentStatus>AUTHORIZED</c:paymentStatus>
    <c:processorTransactionID>4KP97418B91840630</c:processorTransactionID>
  </c:apCheckStatusReply>
</c:replyMessage>
```

Sale Service

Example 28 Check Status Request: Sale Service

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <apPaymentType>PPL</apPaymentType>
  <apCheckStatusService run="true">
    <CheckStatusRequestID>1124439821106016799952</checkStatusRequestID>
  </apCheckStatusService>
</requestMessage>
```

Example 29 Check Status Reply: Sale Service

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>500722420177946002301009</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:apCheckStatusReply>
    <c:reasonCode>100</c:reasonCode>
    <c:reconciliationID>0ELJXCHZ00000000000000000000J0BLF9SV</c:reconciliationID>
    <c:paymentStatus>SETTLED</c:paymentStatus>
    <c:processorTransactionID>7LE62945SB636124F</c:processorTransactionID>
  </c:apCheckStatusReply>
</c:replyMessage>
```

Refund Service

Example 30 Check Status Request: Refund Service

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <apPaymentType>PPL</apPaymentType>
  <apCheckStatusService run="true">
    <checkStatusRequestID>48961921504567098712009999</checkStatusRequestID>
  </apCheckStatusService>
</requestMessage>
```

Example 31 Check Status Reply: Refund Service

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>7649816002777888</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:apCheckStatusReply>
    <c:reasonCode>100</c:reasonCode>
    <c:reconciliationID>ZZEWULIG00000000000000000000IZWESAOC</c:reconciliationID>
    <c:paymentStatus>REFUNDED</c:paymentStatus>
    <c:processorTransactionID>8FU4435670486773H</c:processorTransactionID>
  </c:apCheckStatusReply>
</c:replyMessage>
```

Billing Agreement Service

Example 32 Check Status Request: Billing Agreement Service

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.138">
  <merchantID>mid123</merchantID>
  <merchantReferenceCode>1234</merchantReferenceCode>
  <apPaymentType>PPL</apPaymentType>
  <apCheckStatusService run="true"/>
    <sessionsRequestID>4955835535216022201540</sessionsRequestID>
  <ap>
    <billingAgreementID>SOK-3FJ6025</billingAgreementID>
  </ap>
</requestMessage>
```

Example 33 Check Status Reply: Billing Agreement Service

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.138">
  <c:merchantReferenceCode>1234</c:merchantReferenceCode>
  <c:requestID>4926430141876011901200</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:apCheckStatusReply>
    <c:reasonCode>100</c:reasonCode>
    <c:paymentStatus>ACTIVE</c:paymentStatus>
    <c:processorTransactionID>B-4DR69719PY986990T</c:processorTransactionID>
  </c:apCheckStatusReply>
  <c:apReply>
    <c:payerID>3N53BA8B3MCU4</c:payerID>
  </c:apReply>
  <c:shipTo>
    <c:firstName>Jane</c:firstName>
    <c:lastName>Smith</c:lastName>
    <c:street1>123 Main Street</c:street1>
    <c:street2>Mailstop A</c:street2>
    <c:city>Small Town</c:city>
    <c:state>CA</c:state>
    <c:country>US</c:country>
    <c:postalCode>98765</c:postalCode>
  </c:shipTo>
  <c:billTo>
    <c:firstName>Jane</c:firstName>
    <c:lastName>Smith</c:lastName>
    <c:street1>123 Main St</c:street1>
    <c:city>Small Town</c:city>
    <c:state>CA</c:state>
    <c:country>US</c:country>
    <c:postalCode>98765</c:postalCode>
    <c:email>jsmith@example.com</c:email>
  </c:billTo>
</c:replyMessage>
```

Reason Codes

[Table 11](#) describes the reason codes returned by the Simple Order API for the PayPal services. For a description of replies, decisions, and reason codes, and information about handling replies, see [Getting Started with CyberSource Advanced for the Simple Order API](#).

Because CyberSource can add reply fields and reason codes at any time:

- You must parse the reply data according to the names of the fields instead of the field order in the reply. For more information about parsing reply fields, see the documentation for your client.
 - Your error handler should be able to process new reason codes without problems.
 - Your error handler should use the **decision** field to determine the result if it receives a reason code that it does not recognize.
-

For a description of each processor response, see:

<https://developer.paypal.com/docs/api/payments/#errors>

Table 11 Reason Codes and Processor Responses

Reason Code	Description	Processor Responses
100	Successful transaction.	
150	General system failure. See your CyberSource client documentation for information about handling retries in the case of system errors.	DATA_RETRIEVAL DUPLICATE_REQUEST_ID INTERNAL_SERVICE_ERROR PAYMENT_REQUEST_ID_INVALID
202	Expired card. You might also receive this value if the expiration date you provided does not match the date the issuing bank has on file. Possible action: Request a different card or other form of payment.	EXPIRED_CREDIT_CARD

Table 11 Reason Codes and Processor Responses

Reason Code	Description	Processor Responses
203	<p>General decline of the card. No other information was provided by the issuing bank.</p> <p>Possible action: Request a different card or other form of payment.</p>	<p>CREDIT_CARD_CVV_CHECK_FAILED</p> <p>CREDIT_CARD_REFUSED</p>
223	<p>Possible reasons:</p> <ul style="list-style-type: none"> ■ Merchant account or customer's account is not set up to process such transactions. ■ Insufficient funds in the customer's funding source associated with the account, or transaction declined by bank. ■ A particular action is not permitted, for example: capture refused, or the authorization has already been captured. ■ Fraud setting for the seller is blocking such payments. ■ Payment approval by the customer has expired, and the merchant must restart the payment flow starting from payment creation and direct the customer back to PayPal. <p>Important If an order or authorization returns INSTRUMENT_DECLINED, direct the customer to their PayPal account to choose a different funding option and resubmit the order. Possible reasons for the INSTRUMENT_DECLINED response can include:</p> <ul style="list-style-type: none"> ■ The billing address associated with the financial instrument could not be confirmed. ■ The transaction exceeds the card limit. ■ The transaction was denied by the card issuer. 	<p>BUYER_NOT_SET</p> <p>CANNOT_PAY_SELF</p> <p>CANNOT_REAUTH_CHILD_AUTHORIZATION</p> <p>CANNOT_REAUTH_INSIDE_HONOR_PERIOD</p> <p>CREDIT_PAYMENT_NOT_ALLOWED</p> <p>EXPIRED_CREDIT_CARD_TOKEN</p> <p>FAILED_TO_CHARGE_CC</p> <p>FEATURE_UNSUPPORTED_FOR_PAYEE</p> <p>FULL_REFUND_NOT_ALLOWED_AFTER_PARTIAL_REFUND</p> <p>IMMEDIATE_PAY_NOT_SUPPORTED</p> <p>INSTRUMENT_DECLINED</p> <p>INSUFFICIENT_FUNDS</p> <p>INVALID_FACILITATOR_CONFIGURATION</p> <p>MAXIMUM_ALLOWED_AUTHORIZATION_REACHED_FOR_ORDER</p> <p>MERCHANT_NOT_ENABLED_FOR_CHANNEL_INITIATED_BILLING</p> <p>MERCHANT_NOT_ENABLED_FOR_REFERENCE_TRANSACTION</p> <p>NEED_CREDIT_CARD</p> <p>NEED_CREDIT_CARD_OR_BANK_ACCOUNT</p> <p>NOT_IMPLEMENTED</p> <p>NO_EXTERNAL_FUNDING_DETAILS_FOUND</p> <p>ORDER_ALREADY_COMPLETED</p> <p>ORDER_VOIDED</p>

Table 11 Reason Codes and Processor Responses

Reason Code	Description	Processor Responses
		PAYEE_ACCOUNT_LOCKED_OR_CLOSED
		PAYEE_ACCOUNT_NO_CONFIRMED_EMAIL
		PAYEE_ACCOUNT_RESTRICTED
		PAYEE_BLOCKED_TRANSACTION
		PAYEE_COUNTRY_NOT_ENABLED
		PAYER_ACCOUNT_RESTRICTED
		PAYER_ACTION_REQUIRED
		PAYER_AUTHENTICATION_REQUIRED
		PAYER_CANNOT_PAY
		PAYER_COUNTRY_NOT_ENABLED
		PAYMENT_ALREADY_DONE
		PAYMENT_APPROVAL_EXPIRED
		PAYMENT_CANNOT_BE_INITIATED
		PAYMENT_DENIED
		PAYMENT_EXPIRED
		PAYMENT_METHOD_UNUSABLE
		PAYMENT_NOT_APPROVED_FOR_EXECUTION
		PAYMENT_STATE_INVALID
		REDIRECT_PAYER_FOR_ALTERNATE_FUNDING
		REFUND_EXCEEDED_TRANSACTION_AMOUNT
		REFUND_TIME_LIMIT_EXCEEDED
		REQUIRED_SCOPE_MISSING
		SENDING_LIMIT_EXCEEDED
		TOO_MANY_REAUTHORIZATIONS
		TRANSACTION_ALREADY_REFUNDED
		TRANSACTION_LIMIT_EXCEEDED
		TRANSACTION_REFUSED
		TRANSACTION_REFUSED_BY_PAYPAL_RISK
		TRANSACTION_REFUSED_PAYEE_PREFERENCE
		UNSUPPORTED_PAYEE_COUNTRY

Table 11 Reason Codes and Processor Responses

Reason Code	Description	Processor Responses
		UNSUPPORTED_PAYEE_CURRENCY UNSUPPORTED_SEPA_BANK
231	Invalid account number. Possible action: Request a different card or other form of payment.	BANK_ACCOUNT_VALIDATION_FAILED
233	General decline by the processor.	AMOUNT_MISMATCH AGREEMENT_ALREADY_CANCELLED BATOKEN_MISMATCH BANK_MRN_MISMATCH BUYER_COUNTRY_NOT_ENABLED CANNOT_PAY_THIS_PAYEE CURRENCY_MISMATCH CURRENCY_NOT_ALLOWED EXECUTE_AGREEMENT_BUYER_NOT_ACCEPTED EXECUTE_AGREEMENT_ALREADY_CREATED EXECUTE_AGREEMENT_DOES_NOT_OWN_TOKEN INVALID_ARGUMENT INVALID_CITY_STATE_ZIP INVALID_EXPERIENCE_PROFILE_ID INVALID_PAYER_ID INVALID_PAYMENT_METHOD MERCHANT_COUNTRY_NOT_ENABLED REFUSED_MARK_REF_TXN_NOT_ENABLED PAYEE_FILTER_RESTRICTIONS PAYER_CANNOT_PAY PAYER_INFO_NULL PAYER_EMPTY_BILLING_ADDRESS PAYER_ID_MISSING_FOR_CARD_TOKEN PHONE_NUMBER_REQUIRED REDIRECT_REQUIRED

Table 11 Reason Codes and Processor Responses

Reason Code	Description	Processor Responses
		RT_AGREEMENT_ALREADY_ CANCELED SELECTED_PLAN_NOT_AVAILABLE SHIPPING_ADDRESS_INVALID UNABLE_TO_COMPLETE_ TRANSACTION
235	The requested capture amount exceeds the originally authorized amount. Possible action: Issue a new authorization and capture request for the new amount.	CAPTURE_AMOUNT_LIMIT_ EXCEEDED
237	The authorization has already been reversed. Possible action: No action required.	AUTHORIZATION_VOIDED
242	You requested a capture, but there is no corresponding, unused authorization record. Occurs if there was not a previously successful authorization request or if the previously successful authorization has already been used by another capture request. Possible action: Request a new authorization, and if successful, proceed with the capture.	AUTHORIZATION_EXPIRED AUTHORIZATION_ID_DOES_NOT_ EXIST
246	One of the following: <ul style="list-style-type: none"> ■ The capture or credit is not voidable because the capture or credit information has already been submitted to your processor. ■ You requested a void for a type of transaction that cannot be voided. Possible action: No action required.	AUTHORIZATION_CANNOT_BE_ VOIDED ORDER_CANNOT_BE_VOIDED