



Website Payments Pro Integration Guide

PayPal Website Payments Pro Integration Guide

Document Number: 100001.en_US-200802

© 2008 PayPal, Inc. All rights reserved. PayPal is a registered trademark of PayPal, Inc. The PayPal logo is a trademark of PayPal, Inc. Other trademarks and brands are the property of their respective owners.
The information in this document belongs to PayPal, Inc. It may not be used, reproduced or disclosed without the written approval of PayPal, Inc. PayPal Europe S.à.r.l. & Cie, S.C.A. is authorized and regulated by the Commission de Surveillance du Secteur Financier in Luxembourg as a bank. PayPal RCS registration number: 118349

Notice of non-liability:

PayPal, Inc. is providing the information in this document to you "AS-IS" with all faults. PayPal, Inc. makes no warranties of any kind (whether express, implied or statutory) with respect to the information contained herein. PayPal, Inc. assumes no liability for damages (whether direct or indirect), caused by errors or omissions, or resulting from the use of this document or the information contained in this document or resulting from the application or use of the product or service described herein. PayPal, Inc. reserves the right to make changes to any information herein without further notice.



Contents

Preface	7
This Document	7
Intended Audience	7
Revision History	7
Notational Conventions	8
Chapter 1 Website Payments Pro Overview	11
How Website Payments Pro Works	11
Direct Payment API Overview	12
Express Checkout Overview	12
How It Works	13
Reference Transaction Processing	13
Website Payments Pro Business Rules	13
Compatibility	14
Funding Sources	14
Get Started Quickly: Integration Center	14
Chapter 2 Button Placement, Page Designs, and Programming Flow .15	
HTML for PayPal Button Graphics	15
Examples of Button Placement	16
Relation of Button to API Call: SetExpressCheckout and Redirect to PayPal	17
Design Variation: Eliminating Your Order Review	17
Payment Method Page Layout Recommendations	18
Page Behavior When PayPal Is Selected	19
Express Checkout Programming Flow-of-Control	20
Chapter 3 How Express Checkout Works	23
Relationship to Authorization & Capture	25
Step 1a: Customer Selects PayPal on Your Website	25
Step 1b: Integration Point 1	26
Usage Notes About SetExpressCheckout Elements	27

Transferring Your Customer to PayPal	29
Step 2a: Customer Approves Use of PayPal.	29
Step 2b: Customer Returns to Your Website.	32
Step 2c: Integration Point 2	32
ReturnURL, CancelURL, and the Express Checkout Token	33
Usage Notes About GetExpressCheckoutDetails Elements.	33
Step 3a: Customer Completes Order	34
Step 3b: Integration Point 3	36
Usage Notes About DoExpressCheckoutPaymentRequest Elements	37
Step 4: Customer Notified Order Is Complete	39
Chapter 4 How Direct Payment API Works	41
Technical Overview	41
Relationship to Authorization & Capture	42
Chapter 5 Recurring Payments	43
Overview	43
How Recurring Payments Work	43
Limitations	44
Recurring Payments Terms	44
Recurring Payments With Express Checkout	45
Overview.	45
Recurring Payments With Direct Payments	49
Options for Creating a Recurring Payments Profile	50
Specifying the Regular Payment Period	50
Including an Optional Trial Period	51
Specifying an Initial Payment	52
Other Profile Options	52
Recurring Payments Profile Status.	53
Getting Recurring Payments Profile Information.	53
Modifying a Recurring Payments Profile	54
Updating Addresses	54
Updating the Billing Amount	55
Billing the Outstanding Amount of a Profile	55
PayPal Notifications	55
Chapter 6 How Authorization & Capture Works	57

Fundamental Authorization Process With the APIs	57
Honor Period and Authorization Period	58
Order Authorizations Scenarios	59
Simple Order.	59
Complex Order.	59
Concurrent Authorizations	60
Total Capture Hits Relative Tolerance With Open Authorizations	61
Void Authorizations	62
Partial Capture.	62
Complete Capture	63
Optimal Buyer Experience	64
Capturing Funds on Basic Authorizations	64
Buyer Approval for Basic Authorizations.	64
Voiding Basic Authorizations.	64
Chapter 7 Integrating giropay with Express Checkout	67
Processing Page Flow	67
Giropay Payment Page Flow	67
Cancelled or Unsuccessful Giropay Payment Page Flow	68
Giropay Integration.	68
Initiate the Flow with SetExpressCheckout	69
Redirect the Customer to PayPal	69
Complete the Transaction	70
Receive Transaction Status Notification	70



Preface

This Document

This document describes PayPal Direct Payment and PayPal Express Checkout as payment solutions for customer checkout on your website.

Intended Audience

This document is written for merchants who use either PayPal Direct Payment or PayPal Express Checkout and the programmers who implement these products on a merchant's website.

Revision History

Revision history for *PayPal Website Payments Pro Integration Guide*.

TABLE P.1 Revision History

Date	Description
February 2008	Added that API version must be 50.0 for recurring payments.
January 2008	Added recurring payments chapter.
September 2007	Added chapter on the giropay funding method.
May 2007	Added information about reference transactions and moved DoReferenceTransaction API to the Name-Value Pair API Developer Guide and Reference and the SOAP API Developer Reference .
March 2007	Added chapter about the DoReferenceTransaction API
February 2007	Minor corrections.
December 2006	New button placement requirements. New PayPal Checkout button graphic.
September 2006	<ul style="list-style-type: none">• Description of the useraction variable that can be used on the redirection of the user's browser to PayPal after SetExpressCheckout to control the text of the final button displayed on the PayPal site.• All information about the SOAP APIs for Express Checkout and Authorization & Capture has been moved to the SOAP API Developer Reference.
August 2006	Minor corrections

TABLE P.1 Revision History

Date	Description
July 2006	Direct Payment API now supports the Switch and Solo credit cards and can be used with any PayPal-supported currency.
May 2006	Miscellaneous updates
March 2006	Miscellaneous corrections
January 2006	Additional API error messages for Express Checkout: 10445, 10446.
December 2005	Removed erroneous description that stated that the SetExpressCheckoutRequest field <code>cpp-header-image</code> must be URL-encoded.

Notational Conventions

This document uses typefaces to identify the characteristics of text. These typefaces and the characteristics they imply are described below:

Typeface	How Used
<i>serif italics</i>	<p>A document title.</p> <p>A term being discussed or defined. For example: A file is a readable or writable stream of characters ...</p> <p>Boolean values (not keywords). For example: The function returns true if it encounters an error.</p>
monospaced	<p>Pathnames or file names that appear in body text frames.</p> <p>Code-related names that appear in body text frames. Such names are used for functions, callbacks, arguments, data structures, and fields. For example: <code>AbstractResponseType</code> is the SOAP response type definition on which all PayPal API response methods are based.</p> <p>Components of Internet protocol requests and responses, such as HTTPS and FORM variables. For example: The PayPal system uses a <code>method=POST</code> request to return IPN status variables related to subscriptions, such as <code>txn_type</code>.</p>
Serif bold	<p>User interface names, such as window names or menu selections. For example: On the Profile page, click Email to confirm your email address.</p>

Typeface	How Used
<i>San-serif</i> <i>oblique</i>	Placeholders used in the context of a format or programming standard or formal descriptions of PayPal system syntax. Placeholders indicate values or names that the reader should provide. Example: For example, amount is the variable for a single-item shopping cart, but amount_ <i>X</i> is the name of the variable for a multi-item shopping cart. amount_ <i>3</i> is the item amount for the third item in a multiple-item shopping cart.

To convey additional information, this document may also apply color and underlining to words or phrases that use the typefaces described above. Such use is described below:

Text attribute	How Used
xxxxxx	Hypertext link to a page in the current document or to another document in the set.
xxxxxx	Hypertext link to a URL or that initiates a web action, such as sending mail.



1

Website Payments Pro Overview

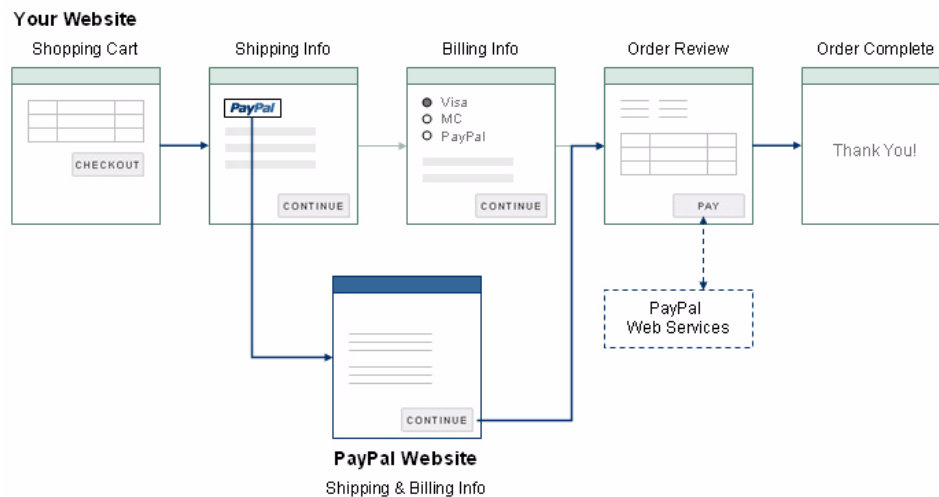
With Website Payments Pro, you get the payment processing capabilities of a merchant account and gateway – plus much more. It is an all-in-one payment solution that includes PayPal Direct Payment API and PayPal Express Checkout.

- **Direct Payment API** enables you to accept credit card payments directly on your website. PayPal remains invisible, so you control the customer experience. You can also use reference transactions to implement recurring payments or modify a previous transaction.
- **PayPal Express Checkout** allows PayPal account holders to check out fast with saved information, and enables you to gain incremental sales from PayPal’s growing base of users.

How Website Payments Pro Works

Figure 1.1, “High-Level View of Website Payments Pro,” is an example of a standard checkout process. Website Payments Pro has the flexibility to work with your unique checkout process, whether it is one page or has multiple steps.

FIGURE 1.1 High-Level View of Website Payments Pro



1. After selecting products to purchase, your customer chooses whether they want to pay using PayPal or pay with credit cards directly on your website.
2. If your customer pays using credit cards on your website, PayPal processes them in the background.

3. If your customer chooses to use PayPal, he is transferred to PayPal to login and select a shipping address and payment method, and is returned to your website to complete his purchase.
4. Once the buyer completes their order, you receive your payment in seconds.

Direct Payment API Overview

The Direct Payment API offers you direct credit card payment processing capability through PayPal. For credit card transactions, customers can stay on your website as PayPal processes the payment in the background.

For each payment, Direct Payment API takes the billing address, transaction amount, credit card information, and item information as inputs. Within seconds, the API returns a confirmation that the transaction has been processed. Additionally, Direct Payment API lets you flag potentially fraudulent transactions, and provides you with industry-standard AVS and CVV2 responses for each transaction.

By integrating Direct Payment API with Express Checkout as part of the Website Payments Pro solution, you can accept all major payment types, including PayPal, while working with a single provider that processes and manages all of your online payments for you.

IMPORTANT: Direct Payment API is not a standalone product. You are required to use Direct Payment API and Express Checkout together as part of the Website Payments Pro solution. See [“Website Payments Pro Business Rules” on page 13](#).

The Direct Payment API is not covered by the PayPal Seller Protection Policy (SPP).

Express Checkout Overview

The more convenient it is for your customers to buy from you, the more they'll buy. Express Checkout allows customers the option to pay quickly through PayPal – and gives your business more benefits:

- **Give buyers more convenience, and get more sales.** Since your customers simply log in to use information they've already entered with PayPal, they save time by completing transactions in fewer steps. This helps increase loyalty and sales.
- **Complete sales on your website, and get more upsell opportunities.** Buyers finish their orders on your website. This gives you more advertising opportunities.
- **Help customers feel safer, so they buy more.** Buyers prefer to pay with PayPal because their customer information is kept safe. When they're confident about the security of their information, they purchase more.

How It Works

1. After selecting products to purchase, your customers click **Checkout with PayPal** on your website.
2. They're transferred to PayPal, where they select their payment method, as well as the correct shipping and billing address, then are returned to your website to complete their purchase.
3. PayPal automatically gives you the shipping address, email address, and other customer information to fulfill the order.

With Express Checkout, your buyers finish their orders on your website, not PayPal's, so you can:

- Get real time notification of successful payments.
- Automate your internal business processes.
- Ensure buyers make it to your final confirmation page.
- Be notified that the buyer's address is confirmed, and ensure you're eligible for coverage under PayPal's Seller Protection Policy.

Reference Transaction Processing

A reference transaction is a financial transaction from which subsequent transactions can be derived. Typically, you use reference transactions to

- make an adjustment to a previous transaction
- re-bill a customer based on a previous transaction
- implement recurring payments

For example, a buyer can make a purchase on your site and the credit card charge, identified by the `DoDirectPayment` API's `TransactionID`, can later be used to initiate another transaction.

Sale and Authorization transactions can make use of a reference transaction as a source of transaction data. The `DoDirectPayment` API's `TransactionID` becomes the *reference transaction ID* in the new transaction.

To implement a reference transaction, use the `DoReferenceTransaction` API, which is described in the [Name-Value Pair API Developer Guide and Reference](#) and the [SOAP API Developer Reference](#).

Website Payments Pro Business Rules

Website Payments Pro must be integrated on your website in the following ways. You should:

1. Present the PayPal Express Checkout button and associated messaging before requesting shipping address, billing address, and financial information. PayPal account holders should not be required enter any of this information on your website, because the information is available from their PayPal accounts.
2. Display PayPal as an option along side other payment methods, wherever other payment methods are offered.
3. Present the PayPal mark graphic wherever other payment marks are displayed.

NOTE: See [Chapter 2, “Button Placement, Page Designs, and Programming Flow”](#) for precise details.

Compatibility

Website Payments Pro works with many other PayPal products, such as Instant Payment Notification, Settlement System, Downloadable History Log, Authorization & Capture, and more.

Funding Sources

With Express Checkout, you can accept all major credit and debit cards, bank transfers, and PayPal balance payments.

With PayPal Direct Payment API, you can accept all major credit and debit cards, including Visa, Master Card, American Express, and Discover.

Get Started Quickly: Integration Center

PayPal’s Integration Center at <https://www.paypal.com/integration> has step-by-step details for getting started with the PayPal Software Development Kits (SDKs), Website Payments Pro, Express Checkout, Website Payments Standard, Authorization & Capture, Instant Payment Notification, and more.

Visit the Integration Center at: <https://www.paypal.com/integration>.



2

Button Placement, Page Designs, and Programming Flow

When you offer PayPal Express Checkout to your customers, you are required to display it in two forms, for your customers' best buying experience:

- PayPal as a Checkout Choice on your shopping cart page
- PayPal as a Payment Method

TABLE 2.1 PayPal Button Placement and Rules

Placement	PayPal Graphic	Requirements
PayPal as a Checkout Choice		Place the PayPal Checkout button on your cart page, aligned with any other checkout buttons.
PayPal as a Payment Method		Place the PayPal Acceptance Mark graphic: 1. On your Payment Method page. 2. On your home page, along with credit card logos, if applicable.

HTML for PayPal Button Graphics

You can get HTML for the Express Checkout button and PayPal Acceptance Mark from the following location:

TABLE 2.2 Location to get HTML for Express Checkout

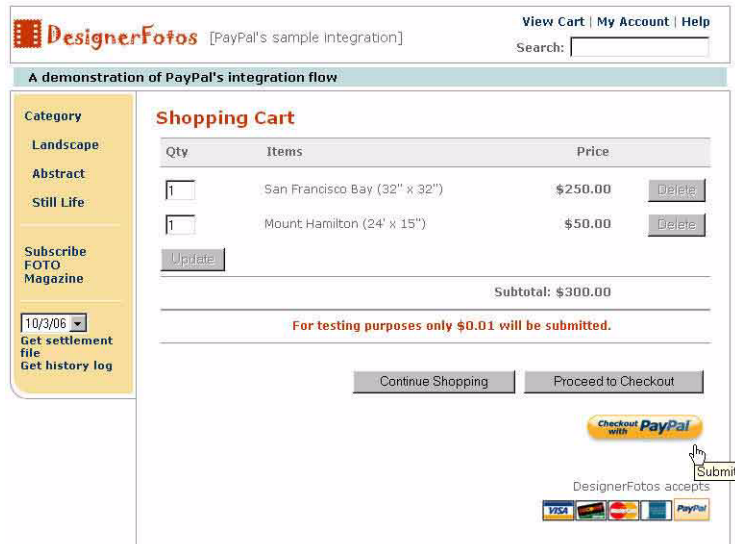
Country	URL
USA	https://www.paypal.com/express-checkout-buttons
United Kingdom	https://www.paypal.com/uk/ec-buttons
Australia	https://www.paypal.com/au/ec-buttons
Other countries	https://www.paypal.com/row/ec-buttons

IMPORTANT: Rather than storing the button graphics on your own server, use the PayPal-provided image paths for the graphics to reassure your customers that the checkout is secure and that you are displaying the most up-to-date logos from PayPal.

Examples of Button Placement

Here are some examples of proper placement:

1. PayPal as a Checkout Choice and as a Payment Method



2. PayPal as a Payment Method



You can choose from several design variations for PayPal as a Payment Method. See [“Payment Method Page Layout Recommendations.”](#)

Relation of Button to API Call: SetExpressCheckout and Redirect to PayPal

Both button graphics must make a call to the SetExpressCheckout API. See the [PayPal SOAP API Reference](#) or the [PayPal NVP API Developer Guide](#) for full programming details about SetExpressCheckout.

After the response from SetExpressCheckout, you must redirect the customer's browser to PayPal. The SetExpressCheckout response includes an Express Checkout session token. Add the value of the Token from the SetExpressCheckout response as a name/value pair to the following URL, and redirect your customer's browser to it:

```
https://www.paypal.com/cgi-bin/webscr?cmd=_express-checkout&token=value_from_SetExpressCheckoutResponse
```

NOTE: Express Checkout has a variation on this redirect URL (called “user action”) that allows you to bypass calling the second API (GetExpressCheckoutDetails) and to change the text of the final button displayed on PayPal. See “[Design Variation: Eliminating Your Order Review](#)” on page 17.

Recommendation for Browser Redirection

For redirecting the customer's browser to the PayPal URL, PayPal recommends that you use the HTTPS response 302 “Object Moved” with the PayPal URL as the value of the Location header in the HTTPS response. Ensure that you use an SSL-enabled server to prevent browser warnings about a mix of secure and insecure graphics.

Design Variation: Eliminating Your Order Review

If your normal checkout includes displaying Payment Methods page towards its end, you do not need to display that Payment Method page after the customer returns from PayPal to your site, because it is superfluous. The customer has already selected PayPal to pay you.

You can make the checkout appear to complete on the PayPal site, not your own, and entirely bypass your own order review page. (After the customer returns from the PayPal site, you must call the DoExpressCheckoutPayment API to actually complete the transaction.) For example, if you do not want to display an “Order Review” page on your site after the customer returns from PayPal, you want the button text on PayPal to read **Pay**.

You control the text of the button displayed on the PayPal site with the `useraction` variable on the PayPal URL to which you redirect the customer after SetExpressCheckout:

- If `useraction` is not set or `useraction=continue`: PayPal displays a **Continue Checkout** button on its site.
- `useraction=commit`: PayPal displays a **Pay** button on its site.

Here are the discrete steps for using `useraction`:

1. Get the token from the response from SetExpressCheckout.

The response from `SetExpressCheckout` is the buyer's token. For example, if the value of `ReturnURL` on `SetExpressCheckout` is `https://www.mybiz.com/snagECvalues`, the URL to which PayPal redirects looks like this:

```
https://www.mybiz.com/snagECvalues?token=EC-0W8920957N684880R
```

2. Add the token and the desired `useraction` as a name/value pairs to the following URL, and redirect your customer's browser to it:

```
https://www.paypal.com/cgi-bin/webscr?cmd=_express-checkout&token=valueFrom  
SetExpressCheckoutResponse&useraction=continue_or_commit
```

Payment Method Page Layout Recommendations

When you display the PayPal Acceptance Mark with other payment methods, you have several designs to choose from:

- As a radio button
- As horizontal fields
- In a pulldown SELECT list

IMPORTANT: Do not preselect any payment method. Allow the customer to make a choice without any default.

As a radio button:

FIGURE 2.1 *Payment Methods: PayPal as Unique Choice*



DesignerFotos

Billing Information

Please select a payment method, enter your billing address, then click the 'Continue' button.

Payment Method

PayPal Save time. Checkout securely. Pay without sharing your financial information.

MasterCard **VISA** **AMEX** **DISCOVER**

Credit or Debit Card Information

Credit Card Type:

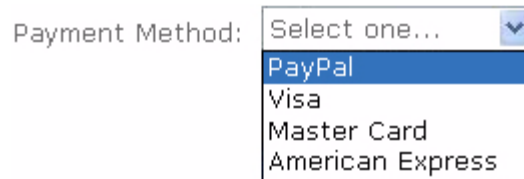
As horizontal option fields:

FIGURE 2.2 Payment Methods: Horizontal Design



As a pulldown SELECT list:

FIGURE 2.3 Payment Methods: Pulldown SELECT List



Page Behavior When PayPal Is Selected

When your customer selects PayPal as a Payment Method, for the best buying experience, you should:

- Use JavaScript to hide or disable credit card fields and billing address fields, because prompting for this information is irrelevant when a customer pays with PayPal.
- If other fields such as coupon code or gift certificate are on the page, still display them so your customer can fill them out before being redirected to PayPal.
- Change the function of the “Continue Checkout” button to call the SetExpressCheckout API and redirect the customer to PayPal.

Express Checkout Programming Flow-of-Control

Your page design triggers PayPal Express Checkout API calls. Here is how the Express Checkout API calls work. For precise details about the fields and values of the Express Checkout API calls, see “Express Checkout API” in the [SOAP API Reference](#).

1. Preparatory indication to PayPal that you intend to use Express Checkout for the payment for the customer’s order: `SetExpressCheckout`.
2. Redirecting the customer to the PayPal website.
3. Getting details from PayPal about this customer (such as shipping address) and optionally allowing the customer to edit these details during the order review: `GetExpressCheckoutDetails`.
4. Making the payment transaction with `DoExpressCheckoutPayment` at the end of checkout, in one of two forms:
 - As the complete amount of a final sale
 - As an authorization for a total amount that you must capture later with `Authorization & Capture`

FIGURE 2.4 Express Checkout Flow-of-Control

PayPal Express Checkout

Legend: Web Flow API Call Token

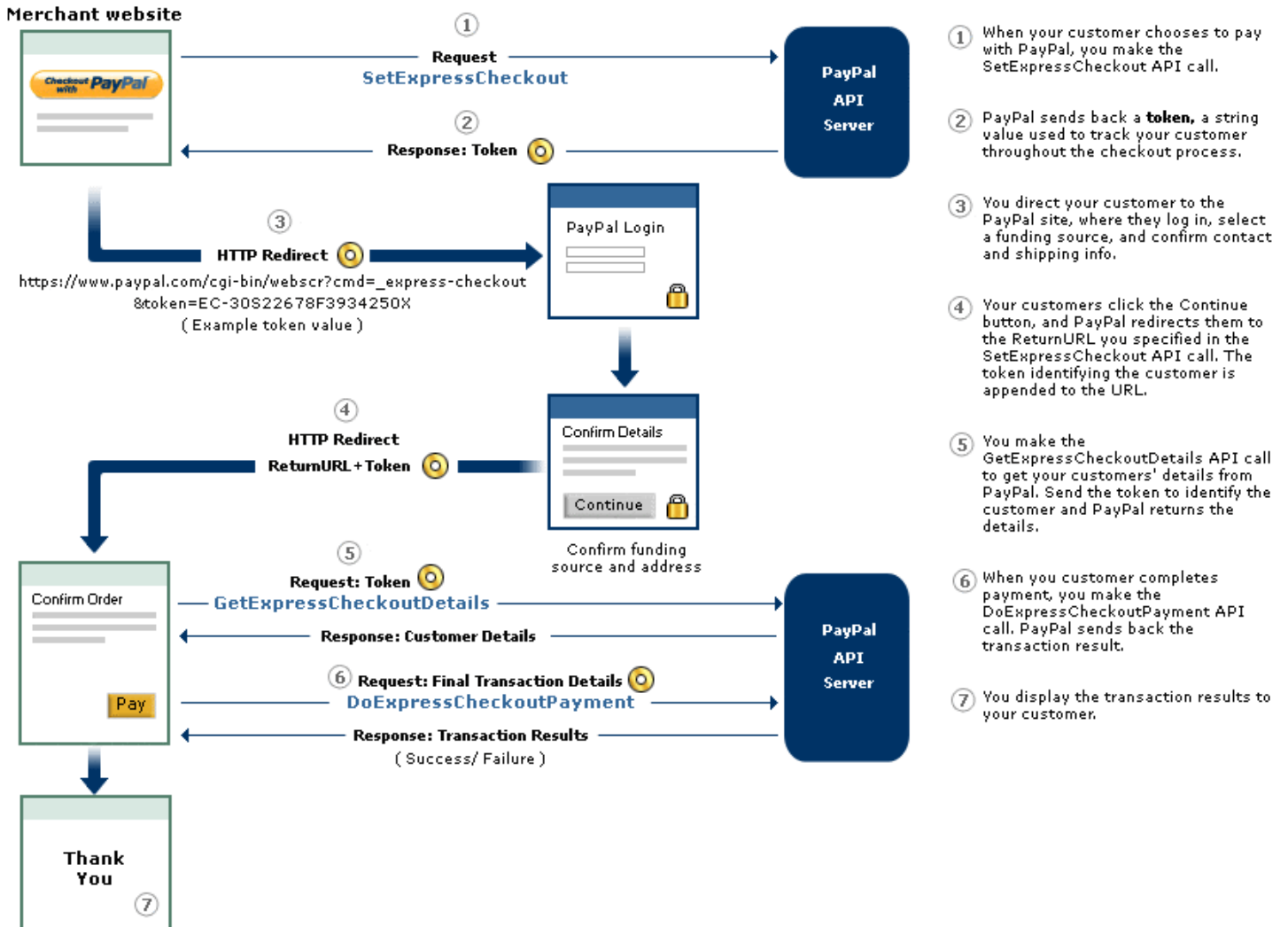


TABLE 2.3 Express Checkout Program Flow-of-Control and Integration Points

Customer...	Merchant...	PayPal...
1 Clicks Checkout with PayPal button.	Calls SetExpressCheckout API with the required fields – estimated OrderTotal, ReturnURL, and CancelURL – and optional fields, such as MaxAmount.	

TABLE 2.3 Express Checkout Program Flow-of-Control and Integration Points

Customer...	Merchant...	PayPal...
		2. Returns <code>SetExpressCheckout</code> response with <code>Token</code> and appends the values of <code>Token</code> and <code>PayerID</code> to your return URL.
	3. Adds value of element <code>Token</code> from <code>SetExpressCheckout</code> response as a name/value pair to the following URL, and redirects the user's browser to it: <code>https://www.paypal.com/cgi-bin/webscr?cmd=_express-checkout&token=value</code>	
Logs in to PayPal, approves the use of PayPal, and clicks Continue or Pay .		4. With a <code>GET</code> , redirects user's browser to merchant's <code>ReturnURL</code> with token value appended.
	5. Optionally calls <code>GetExpressCheckoutDetails</code> API with <code>Token</code> to retrieve customer's information.	Returns <code>GetExpressCheckoutDetails</code> response with <code>PayerID</code> , email address, shipping address, confirmed or unconfirmed status of that shipping address, and other details.
	Renders page in customer's browser for the next step in checkout process, such as your "Order Review" page.	
Clicks "Confirm Order" button	6. Calls <code>DoExpressCheckoutPayment</code> API with the required elements <code>Token</code> , <code>OrderTotal</code> , <code>PaymentAction</code> , and <code>PayerID</code> returned by <code>GetExpressCheckoutDetails</code> response.	Returns payment info with important <code>TransactionID</code> value and other details about the payment.
	7. Displays "Thank You" page.	

3

How Express Checkout Works

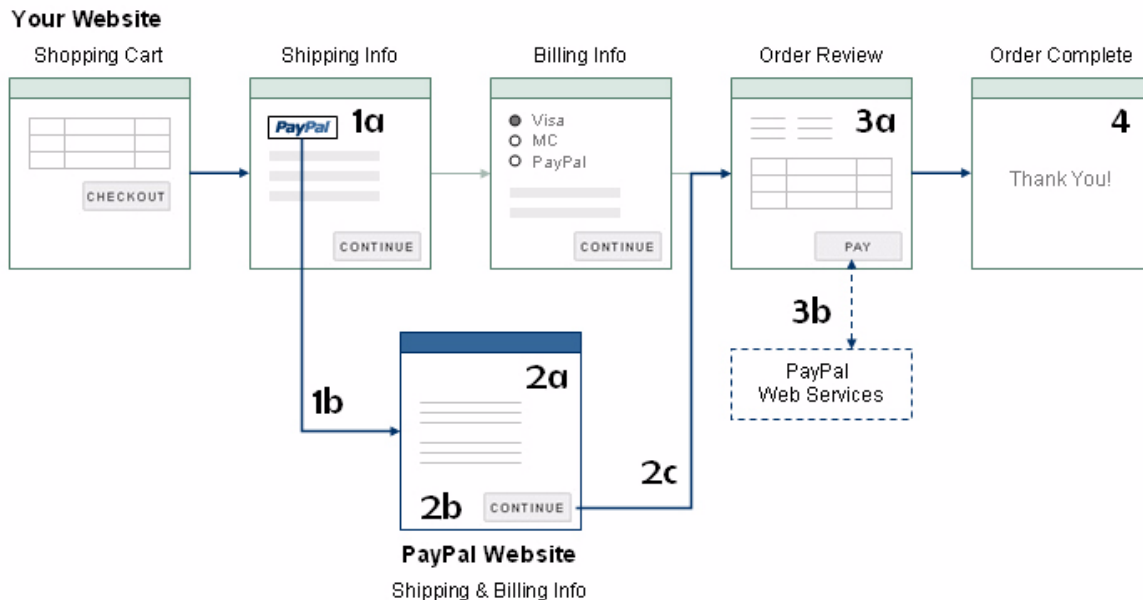
PayPal Express Checkout is a combination of the checkout process on your website, PayPal login and review pages on <https://www.paypal.com>, and PayPal Web Services API SOAP requests/responses.

To explain how Express Checkout works, this chapter presents a generalized checkout process and details how Express Checkout can be implemented with it. This generalized checkout process, which might differ from your own, is shown in [Figure 3.1, “Generalized Customer Checkout.”](#) Each numbered step in the diagram coincides with what this guide refers to as Express Checkout Integration Points 1, 2, and 3.

The Integration Points occur in the following sequence. Your customer always starts and completes his order on your website.

1. When a customer clicks **Checkout with PayPal**, he is transferred to PayPal to log in.
2. The customer then selects a shipping address and payment method and approves the use of PayPal.
3. PayPal then returns the customer to your website to review and finalize the order.

FIGURE 3.1 Generalized Customer Checkout



At each Integration Point, you must set certain required API element values, and you can affect the behavior and usefulness of Express Checkout by setting optional elements.

After a succinct description of the technical view of implementing Express Checkout, the remainder of this chapter includes detailed steps for each of the Integration Points.

TABLE 3.1 Steps in Integrating Express Checkout

Step	Description
1a	<p>After selecting products to purchase, your customer clicks the Checkout with PayPal button on your website.</p> <p>This allows your customer to quickly skip entering shipping and billing information on your website.</p> <p>See “Step 1a: Customer Selects PayPal on Your Website” on page 25.</p>
1b	<p>Integration Point 1</p> <p>You make an API call to pass PayPal the transaction details.</p> <p>You then transfer the customer to PayPal via an HTTP redirect.</p> <p>Your customer is transferred to PayPal.</p> <p>See “Step 1b: Integration Point 1” on page 26.</p>
2a	<p>Your customer selects a shipping address and payment method stored on PayPal.</p> <p>See “Step 2a: Customer Approves Use of PayPal” on page 29.”</p>
2b	<p>Your customer clicks Continue Checkout to approve the use of PayPal and is returned to your website.</p> <p>See “Step 2b: Customer Returns to Your Website” on page 32.</p>
2c	<p>Integration Point 2</p> <p>Your customer is transferred back to your website.</p> <p>PayPal transfers the customer via an HTTP redirect. You then make an API call to retrieve transaction details, such as shipping address, email address, and other information needed to fulfill your order.</p> <p>See “Step 2c: Integration Point 2” on page 32.</p>
3a	<p>Your customer finishes the checkout process on your website, reviews the order, and completes the order.</p> <p>See “Step 3a: Customer Completes Order” on page 34.</p>
3b	<p>Integration Point 3.</p> <p>When your customer places the order, you make an API call to PayPal to request payment. (Your customer does not see this step.)</p> <p>The payment transaction is initiated, and PayPal sends your customer an email receipt for the payment.</p> <p>See “Step 3b: Integration Point 3” on page 36.</p>
4	<p>You transfer your customer to your order confirmation page.</p> <p>See “Step 4: Customer Notified Order Is Complete” on page 39.</p>

Express Checkout gives you the flexibility to put PayPal either first in your checkout process or on your billing page along with other payment options.

Relationship to Authorization & Capture

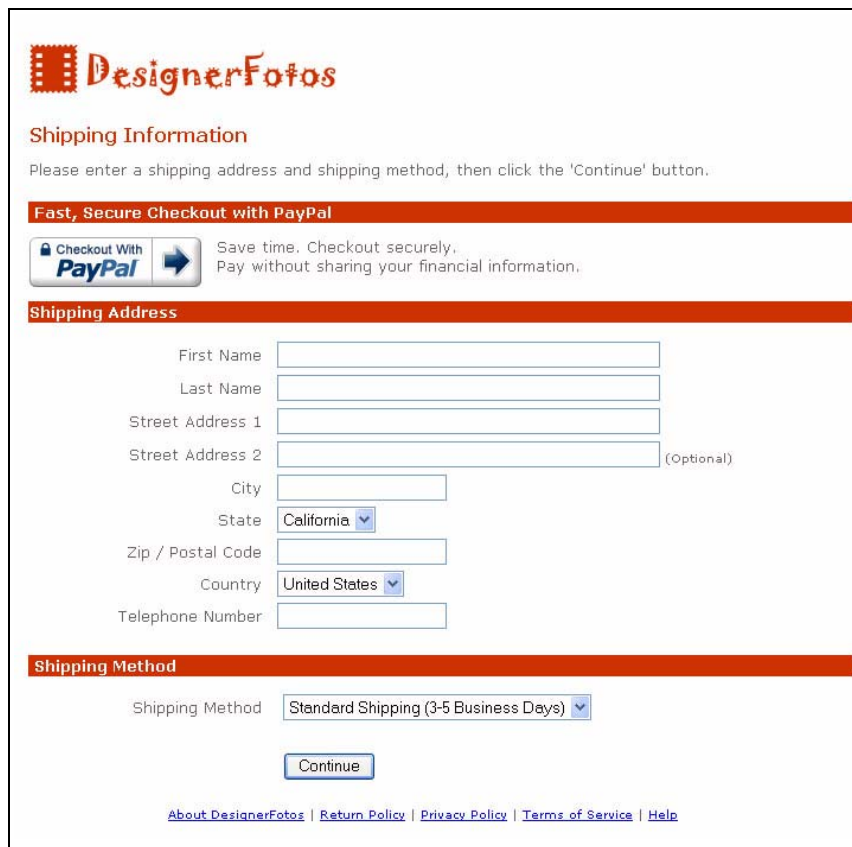
PayPal assumes that at the end of the checkout process, you will make a final sale and payment transaction via PayPal. If at point of sale you do not know the complete cost of the order—for example, if shipping, handling, and tax is not precisely known, or if you want to upsell—you can authorize a transaction that you capture later with Authorization & Capture.

For more information about Authorization & Capture, see [Chapter 6, “How Authorization & Capture Works.”](#)

Step 1a: Customer Selects PayPal on Your Website

PayPal recommends that you place the Express Checkout button on your website before your customers are required to enter their shipping and billing information, as shown in [Figure 3.2, “PayPal Express Checkout Button Before Shipping Address Information.”](#)

FIGURE 3.2 *PayPal Express Checkout Button Before Shipping Address Information*



The screenshot shows the DesignerFotos website checkout page. At the top, the logo "DesignerFotos" is displayed. Below it, the "Shipping Information" section is titled, followed by the instruction: "Please enter a shipping address and shipping method, then click the 'Continue' button." A prominent red banner reads "Fast, Secure Checkout with PayPal". Below this banner is a "Checkout With PayPal" button with a right-pointing arrow. To the right of the button, the text says: "Save time. Checkout securely. Pay without sharing your financial information." The "Shipping Address" section contains several input fields: "First Name", "Last Name", "Street Address 1", "Street Address 2 (Optional)", "City", "State" (a dropdown menu currently showing "California"), "Zip / Postal Code", "Country" (a dropdown menu currently showing "United States"), and "Telephone Number". Below the address fields is the "Shipping Method" section, which includes a dropdown menu currently set to "Standard Shipping (3-5 Business Days)". At the bottom of the form is a "Continue" button. At the very bottom of the page, there are links for "About DesignerFotos", "Return Policy", "Privacy Policy", "Terms of Service", and "Help".

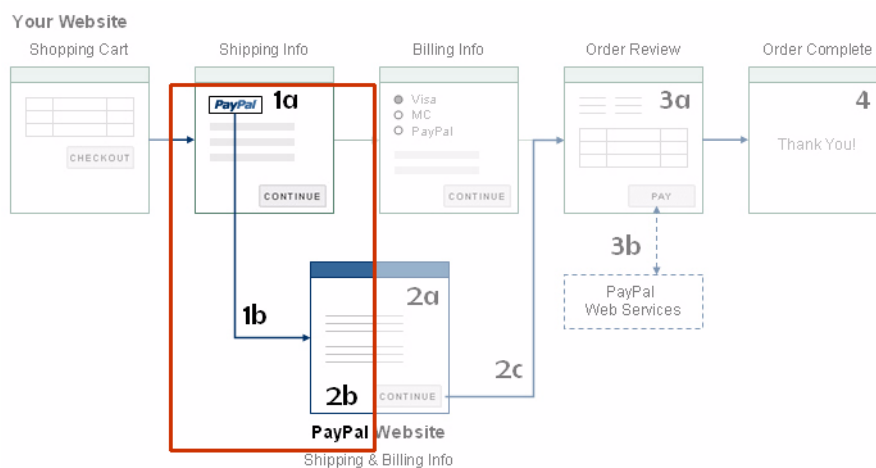
NOTE: Your customer always reviews transaction details and makes the final payment on your website. PayPal handles the payment verification and passes you the customer’s

shipping information. PayPal never shares your customer's financial information with anyone.

Step 1b: Integration Point 1

As shown in [Figure 3.3, "Express Checkout Integration Point 1,"](#) Integration Point 1 is where you transfer a customer's browser to PayPal to select or add a shipping address and funding source.

FIGURE 3.3 Express Checkout Integration Point 1



Integration Point 1 consists of the following events and actions:

- The customer clicks the **Checkout with PayPal** button.
- You send `SetExpressCheckoutRequest` to PayPal.
- PayPal returns the `SetExpressCheckoutResponse`.
- You redirect the customer's browser to PayPal.

Usage Notes About SetExpressCheckout Elements

The following is important usage information about some of the required or optional elements in the first SOAP request for Express Checkout. For complete details about all elements, see the *Name-Value Pair Developer Guide and Reference* or the *SOAP API Reference*.

TABLE 3.2 SetExpressCheckoutRequest Usage Notes

Element	Required or Optional ?	Notes
OrderTotal	Required	<p>The total estimated cost of the order to the customer.</p> <p>If shipping and tax charges are known, include them in OrderTotal; if not, OrderTotal should be the current subtotal of the order.</p>
MaxAmount	Optional	<p>The expected maximum total amount of the complete order, including shipping and tax charges.</p> <p>PayPal uses an adjusted OrderTotal to determine which funding sources it can authorize for use by the customer. PayPal business logic calculations account for the fact that shipping and tax will likely be added to the OrderTotal before the customer completes the purchase. MaxAmount is additional information for PayPal's business logic to properly calculate the customer's available funds for your unique circumstances.</p> <p>If OrderTotal is the final amount, set MaxAmount equal to OrderTotal.</p> <p>NOTE: If the final OrderTotal sent with the DoExpressCheckoutPaymentRequest (the final PayPal Express Checkout API) exceeds the value of MaxAmount, the payment will still be successfully processed.</p>
ReturnURL	Required	<p>URL to which the customer's browser is returned after approving use of PayPal.</p> <p>PayPal recommends that the value of the required ReturnURL element be the final review page on which the customer confirms the order and payment.</p> <p>The value of your ReturnURL must always assume GET as the FORM METHOD, just as if the value were to be included in a FORM. That is, your ReturnURL must expect to read from the QUERY_STRING environment variable, not from standard input.</p> <p>For your programmatic control on the redirect of the customer's browser to your website, the value of ReturnURL can include any name/value pairs your programs require.</p>

TABLE 3.2 SetExpressCheckoutRequest Usage Notes

Element	Required or Optional ?	Notes
CancelURL	Required	<p>URL to which the customer is returned if he decides not to use PayPal or if PayPal is not able to authorize the customer.</p> <p>PayPal recommends that the value of the required CancelURL be the original page on which the customer chose to use PayPal. For instance, if the customer were transferred to PayPal from your shipping information page, the CancelURL value should be the URL of your shipping information page. If the customer clicks Cancel on the PayPal website, PayPal redirects the customer's browser to your shipping information page, where the customer can continue with your standard checkout process.</p>
PaymentAction	Optional	<p>How you want to obtain payment:</p> <ul style="list-style-type: none"> • Sale indicates that this is a final sale for which you are requesting payment. • Authorization or Order indicate that this payment is subject to settlement with PayPal Authorization & Capture. A PaymentAction of Authorization indicates to PayPal that the order total at the end of the customer's checkout will not be a final sale. If you choose this action, you will need to perform an additional step in order to capture the payment. <p>For more information about PayPal Authorization & Capture, see the Chapter 6, "How Authorization & Capture Works."</p>
cpp-header-image	Optional	<p>A URL for the image you want to appear at the top left of the payment page. The image has a maximum size of 750 pixels wide by 90 pixels high.</p> <p>NOTE: PayPal recommends that you provide an image from a secure (https) server. If the image is not on a secure server, when the customer's browser is redirected to the PayPal website, the customer will see a message about potential security risks (a mixture of secure and insecure items). This message might intimidate some customers from continuing with their purchase.</p> <p>In Figure 3.4, "PayPal Login Page" on page 30, the DesignerFotos image beneath the PayPal logo demonstrates how a cpp-header-image appears.</p> <p>For more information about custom payment pages, see the Website Payments Standard Integration Guide.</p>

TABLE 3.2 SetExpressCheckoutRequest Usage Notes

Element	Required or Optional ?	Notes
Custom	Optional	The optional Custom element is a <i>passthrough variable</i> . Its value is returned verbatim on the final PayPal Express Checkout API, DoExpressCheckoutPaymentResponse. You can use this value for whatever purpose you desire, such as an accounting tracking number or additional data needed by your programs (for example, a session-id or other variable).

TABLE 3.3 SetExpressCheckoutResponse Usage Notes

Element	Notes
Token	A timestamped token by which you identify to PayPal that you are processing this payment with Express Checkout. NOTE: The token expires after three hours.

Transferring Your Customer to PayPal

After you receive a successful response from PayPal, you should add the value of the Token from SetExpressCheckoutResponse as a name/value pair to the following URL, and redirect your customer’s browser to it:

```
https://www.paypal.com/cgi-bin/webscr?cmd=_express-checkout&token=value_from_SetExpressCheckoutResponse
```

Express Checkout has a variation on this redirect URL that allows you to bypass calling the second API (GetExpressCheckoutDetails) and to change the text of the final button displayed on PayPal. See “[Design Variation: Eliminating Your Order Review](#)” on page 15.

Recommendation for Browser Redirection

For redirecting the customer’s browser to the PayPal URL, PayPal recommends that you use the HTTPS response 302 “Object Moved” with your URL as the value of the Location header in the HTTPS response. Ensure that you use an SSL-enabled server to prevent browser warnings about a mix of secure and insecure graphics.

Step 2a: Customer Approves Use of PayPal

The next step after redirecting the customer’s browser to PayPal with the response token is for the customer to approve PayPal as the payment method for his purchase. The customer is redirected to the PayPal login page to enter his email and password as shown in [Figure 3.4, “PayPal Login Page.”](#) If the customer does not have a PayPal account, he can click the **Don’t**

How Express Checkout Works

Step 2a: Customer Approves Use of PayPal

have a PayPal account? Click Here link and enter his credit card information to register for a PayPal account.

FIGURE 3.4 PayPal Login Page

DesignerFotos

Payments by
PayPal

Login Review Back to Merchant

Enter Payment Information

PayPal securely processes payments for your merchant, **Designer Fotos**. To continue, please enter the required information below. [Learn more](#) about PayPal.

* indicates required fields [Secure Transaction](#)

PayPal Checkout

If you already have a PayPal account, enter your login details below.

PayPal

*Email Address: [Forgot your email address?](#)

*PayPal Password: [Forgot your password?](#)

Don't have a PayPal account? [Click here](#)

[Cancel and Return to Merchant](#)

If the customer has previously been to PayPal, his email address is pre-filled to save time during login.

After the customer logs in to PayPal, he needs to verify his information on the “Review Your PayPal Information” page, as shown in [Figure 3.5, “PayPal Review Page.”](#)

FIGURE 3.5 PayPal Review Page

The screenshot shows the PayPal Review Page for DesignerFotos. At the top left is the DesignerFotos logo, and at the top right is the Payments by PayPal logo. Below the logo is a progress bar with 'Login', 'Review' (selected), and 'Back to Merchant' options. The main heading is 'Review Your Payment' with a sub-heading 'Review the payment details below and click **Pay** to complete your secure payment.' There is a 'Secure Transaction' lock icon. A note indicates that asterisks mark required fields. The 'Gift Certificates or Coupons' section offers options to 'Select Code' or 'Enter Code' (with a 'Redeem Code' button). Below this is 'Seller Information' (Designer Fotos), 'Shipping Information' (123 Anystreet, Anytown, CA 95123, United States), and 'Payment Method' (Instant Transfer: Fleet Bank XXXXXX4338, Back Up Funding Source: MasterCard XXXX-XXXX-XXXX-1220). At the bottom, there are 'Cancel and Return to Merchant' and 'Pay' buttons.

The customer then:

1. Can review his default funding source and shipping address, select other funding sources or shipping address already saved on PayPal, or enter new ones.

NOTE: PayPal returns your customer to the ReturnURL specified by you in SetExpressCheckoutRequest. If the customer clicks the **Cancel** button, PayPal returns him to the CancelURL specified in the SetExpressCheckoutRequest.

2. Clicks **Pay** to approve the use of PayPal.
3. Returns to your website to complete the purchase.

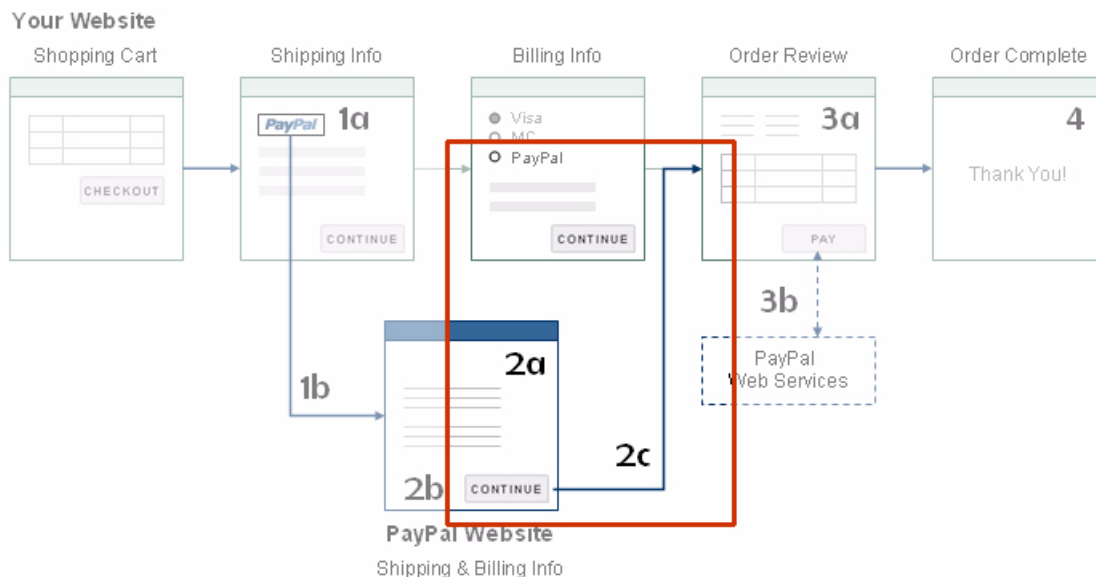
Step 2b: Customer Returns to Your Website

After the customer has selected shipping and billing information on the PayPal website, he clicks **Pay**, which is the customer's approval of the use of PayPal. PayPal then redirects the customer's browser to your website as described in "Step 2c: Integration Point 2."

Step 2c: Integration Point 2

Figure 3.6, "Express Checkout Integration Point 2" illustrates Express Checkout Integration Point 2.

FIGURE 3.6 Express Checkout Integration Point 2



Integration Point 2 consists of the following events and actions:

- The customer clicks **Continue Checkout** on the PayPal review screen.
- PayPal transfers your customer to the location you specified in the ReturnURL.
- You send the `GetExpressCheckoutDetailsRequest` to PayPal.
- PayPal returns the `GetExpressCheckoutDetailsResponse`.
- You display the next screen of your checkout process to your customer.

After your customer has reviewed and approved his financial and shipping information, and done any necessary editing, PayPal redirects his browser to the ReturnURL provided in `SetExpressCheckoutRequest`.

ReturnURL, CancelURL, and the Express Checkout Token

PayPal appends the name/value pairs `token=tokenValue` and `PayerID=payeridValue` to the value of your ReturnURL and CancelURL. For example, if you set ReturnURL as follows:

```
https://www.newco.com/ourcheckout
```

PayPal changes the value as follows:

```
https://www.newco.com/ourcheckout?token=tokenValue&PayerID=payeridValue
```

Similarly, if your ReturnURL value already has name/value pairs, like the following:

```
https://www.newco.com/ourcheckout?cartid=1234
```

PayPal prefixes the appended token with the name/value pair delimiter, like this:

```
https://www.newco.com/ourcheckout?cartid=1234&token=tokenValue&PayerID=payeridValue
```

Once the customer arrives at this ReturnURL, you optionally send the `GetExpressCheckoutDetailsRequest` with the Token value provided in `SetExpressCheckoutResponse`. PayPal then sends you a response with your customer's transaction information.

For a list of all elements in `GetExpressCheckoutDetailsResponse`, see the [SOAP API Reference](#).

Usage Notes About GetExpressCheckoutDetails Elements

The following outlines usage information for some of the important elements sent in this response.

TABLE 3.4 *GetExpressCheckoutDetailsResponse Usage Notes*

Element	Notes
Payer	Email address of the payer.
PayerID	Unique PayPal customer account number. You must provide this value with <code>DoExpressCheckoutPaymentRequest</code> .
PayerStatus	The payer's PayPal account status. A value of <code>Verified</code> means that the customer has confirmed ownership of a bank account or has verified his account status through other means.
FirstName LastName	The payer's name.

TABLE 3.4 *GetExpressCheckoutDetailsResponse Usage Notes*

Element	Notes
Address	<p>The payer's shipping address selected on PayPal.</p> <p>NOTE: With <code>SetExpressCheckoutRequest</code>, if you sent PayPal an <code>Address</code> in and set <code>AddressOverride</code>, on <code>GetExpressCheckoutDetailsResponse</code> PayPal returns the shipping address you originally sent on <code>SetExpressCheckoutRequest</code>.</p>
AddressStatus	<p>The status of the customer's shipping address.</p> <p>A status of <code>Confirmed</code> means that the shipping address matches a billing address on record with PayPal and that billing address has been verified by AVS. Your use of a <code>Confirmed</code> address is one of the requirements for you to be protected by PayPal's Seller Protection Policy (SPP). For more information about the Seller Protection Policy, see the following: https://www.paypal.com/spp</p>
ContactPhone	<p>Payer's contact telephone number.</p> <p><code>ContactPhone</code> is returned to you only if you set this as a preference in the Website Payment Preferences of your Profile on https://www.paypal.com.</p>

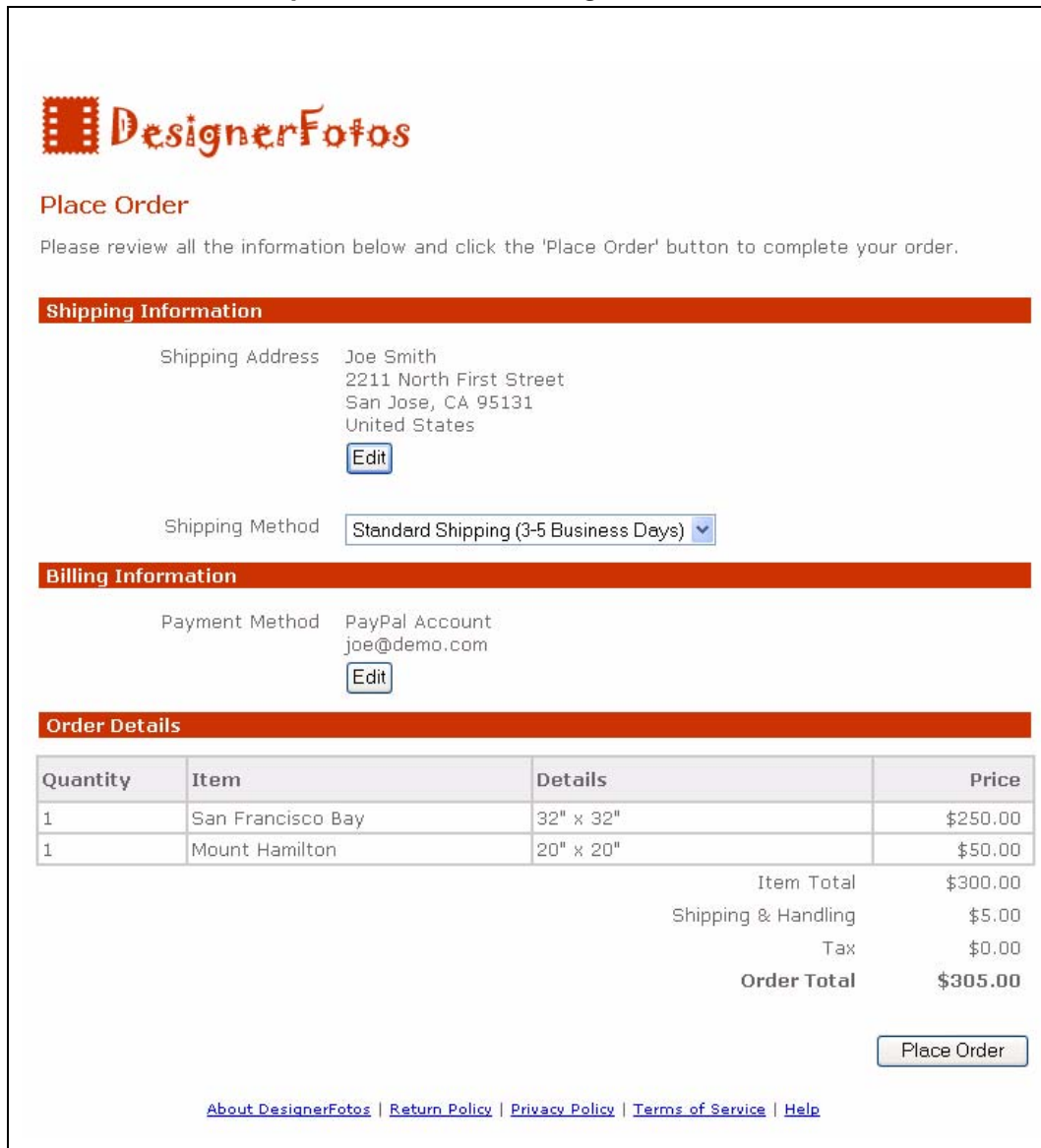
To protect the privacy of your customer, PayPal does not share billing address or financial information, such as credit card numbers.

NOTE: The terms of PayPal's Privacy Policy allow you to use a customer's personal information only for communications relating to the transaction, unless the customer expressly gives you permission to use the information for other purposes. For information about the PayPal Privacy Policy, see <https://www.paypal.com/privacy>.

Step 3a: Customer Completes Order

After you receive a successful `GetExpressCheckoutDetailsResponse`, display the next page in your checkout process. This page might be your order review page or a page on which the customer can select a shipping method, enter shipping instructions, or specify any other information necessary to complete the purchase.

FIGURE 3.7 Example of Order Review Page



DesignerFotos

Place Order

Please review all the information below and click the 'Place Order' button to complete your order.

Shipping Information

Shipping Address: Joe Smith
2211 North First Street
San Jose, CA 95131
United States
[Edit](#)

Shipping Method:

Billing Information

Payment Method: PayPal Account
joe@demo.com
[Edit](#)

Order Details

Quantity	Item	Details	Price
1	San Francisco Bay	32" x 32"	\$250.00
1	Mount Hamilton	20" x 20"	\$50.00
Item Total			\$300.00
Shipping & Handling			\$5.00
Tax			\$0.00
Order Total			\$305.00

[Place Order](#)

[About DesignerFotos](#) | [Return Policy](#) | [Privacy Policy](#) | [Terms of Service](#) | [Help](#)

PayPal recommends that you alter your order review page as follows. Figure 3.7, “Example of Order Review Page” on page 35 is an example of a page that has been altered to reflect these

guidelines.

TABLE 3.5 Shipping, Billing and Order Total Usage

Shipping Information Section	Display the shipping address supplied by PayPal. The “Edit Shipping” button should return your customer to PayPal to edit shipping information on the PayPal website. This allows the customer to quickly select a different address that he already has stored with PayPal, or enter a new address. This also ensures that PayPal can provide you with the updated AddressStatus for the new shipping address. For information about AddressStatus, see the SOAP API Reference .
Billing Information Section	For billing information, display the customer’s PayPal email address provided in Express Checkout.
Order Total	With Express Checkout, you must display to the customer the same exact OrderTotal value that you send to PayPal in DoExpressCheckoutPaymentRequest.

When the customer clicks the “Place Order” button, send DoExpressCheckoutPaymentRequest to initiate the payment. After a successful response is sent from PayPal, direct the customer to your order completion page to inform him that you received his order.

Step 3b: Integration Point 3

Figure 3.8, “Express Checkout Integration Point 3” illustrates Express Checkout Integration Point 3.

FIGURE 3.8 Express Checkout Integration Point 3



Integration Point 3 consists of the following events and actions:

- The customer clicks the “Place Order” button on your website.
- You send the `DoExpressCheckoutPaymentRequest` to PayPal.
- PayPal returns the `DoExpressCheckoutPaymentResponse`.
- You redirect the customer to your “Order Confirmation” page.

Usage Notes About `DoExpressCheckoutPaymentRequest` Elements

Table 3.6 provides important usage information about some of the required or optional elements in the final API request for Express Checkout.

TABLE 3.6 `DoExpressCheckoutPaymentRequest` Usage Notes

Element	Required or Optional?	Notes
<code>OrderTotal</code>	Required	Total of the order, including shipping, handling, and tax. This must be the final amount of the purchase and can differ from the original, estimated <code>OrderTotal</code> you sent in the <code>SetExpressCheckoutRequest</code> . NOTE: PayPal does not enforce a maximum difference between the original estimated <code>OrderTotal</code> and the one you send on <code>DoExpressCheckoutPaymentRequest</code> . However, if the difference is substantial, your customer may no longer be approved to use the funding source he originally accepted.

TABLE 3.6 DoExpressCheckoutPaymentRequest Usage Notes

Element	Required or Optional?	Notes
PaymentAction	Required	<p>How you want to obtain payment:</p> <ul style="list-style-type: none"> • Sale indicates that this is a final sale for which you are requesting payment. • Authorization or Order indicate that this payment is subject to settlement with PayPal Authorization & Capture. <p>NOTE: If you set PaymentAction to Sale or Order on SetExpressCheckoutRequest, you cannot change the PaymentAction value on DoExpressCheckoutPaymentRequest.</p> <p>You can, however, set PaymentAction to Sale on DoExpressCheckoutPaymentRequest if you originally set it to Authorization on SetExpressCheckoutRequest.</p> <p>NOTE: PayPal requires that a merchant using Express Checkout display to the customer the same amount that the merchant sends to PayPal for initial authorization in the OrderTotal element with the DoExpressCheckoutPaymentRequest API.</p>
ItemTotal ShippingTotal HandlingTotal TaxTotal	Optional	<p>You can send individual order-level totals for items, shipping, handling, and tax. If you send this information, PayPal includes it in the customer's transaction receipt from PayPal and on PayPal's Transaction Details page.</p> <p>If you send this information, the sum of these four totals must equal OrderTotal.</p>
Order Description	Optional	<p>The description you want to appear on the customer's transaction receipt from PayPal and on PayPal's Transaction Details page.</p> <p>PayPal recommends that you include your order number for this purchase if one is available.</p>

TABLE 3.6 DoExpressCheckoutPaymentRequest Usage Notes

Element	Required or Optional?	Notes
PaymentDetailsItem	PaymentDetailsItem is optional.	If you send details about each item, they are included in the customer's transaction receipt from PayPal and on PayPal's Transaction Details page.
<ul style="list-style-type: none"> • Name • Number • Amount • Quantity • SalesTax 	PaymentDetailsItem is required.	Providing this information allows your customer to review the purchase information in his PayPal account details, can remind your customer about the purchase details, and might decrease the likelihood that your customer will mistakenly file a chargeback.
PaymentDetails		The following rules apply to the PaymentDetailsItem elements:
<ul style="list-style-type: none"> • ItemTotal • TaxTotal 		<ol style="list-style-type: none"> 1. If you set PaymentDetailsItem.Name, you must also send PaymentItem.Amount. 2. If you set PaymentDetailsItem.Amount, the sum of (PaymentDetailsItem.Amount x PaymentDetailsItem.Quantity) for all payment items must equal PaymentDetails.ItemTotal. 3. If you set PaymentDetailsItem.SalesTax, the sum of (PaymentDetailsItem.SalesTax x PaymentDetailsItem.Quantity) for all payment items must equal PaymentDetails.TaxTotal.
ShipToAddress	Optional	<p>If you allow the customer to enter or edit shipping information on your website, you should pass your customer's shipping address to PayPal.</p> <p>NOTE: If you are using the shipping address PayPal returned to you with GetExpressCheckoutDetailsResponse, do not pass this address back to PayPal on DoExpressCheckoutPaymentRequest.</p> <p>NOTE: PayPal recommends that, whenever possible, you allow your customer to use the shipping address stored by PayPal. Doing so helps your customer complete the checkout more quickly and allows PayPal to provide you with the shipping AddressStatus in GetExpressCheckoutDetailsResponse.</p>

Step 4: Customer Notified Order Is Complete

After the customer approves the order and it completes successfully, PayPal recommends that you display an order completion page showing the details of the transaction. An example is shown below.

FIGURE 3.9 Example of Order Complete Page

DesignerFotos

Your Order is Complete

Please print this receipt for your records.

Shipping Information

Shipping Address Joe Smith
2211 North First Street
San Jose, CA 95131
United States

Shipping Method Standard Shipping (3-5 Business Days)

Billing Information

Payment Method PayPal Account
joe@demo.com

PayPal Transaction ID 823487029384790

Order Details

PhotoPal Order Number 123-456-789

Quantity	Item	Details	Price
1	San Francisco Bay	32" x 32"	\$250.00
1	Mount Hamilton	20" x 20"	\$50.00
Item Total			\$300.00
Shipping & Handling			\$5.00
Tax			\$0.00
Order Total			\$305.00

[About DesignerFotos](#) | [Return Policy](#) | [Privacy Policy](#) | [Terms of Service](#) | [Help](#)

PayPal sends the customer an email notification with the completed transaction details of the order. You also receive an email of the payment and can view it in your Downloadable History Log or Account Overview.

4

How Direct Payment API Works

PayPal Direct Payment API offers you direct credit card payment processing through PayPal. To pay with a credit card, your customer never leaves your website, as PayPal processes the payment behind the scene.

FIGURE 4.1 Generalized Customer Checkout with Direct Payment API

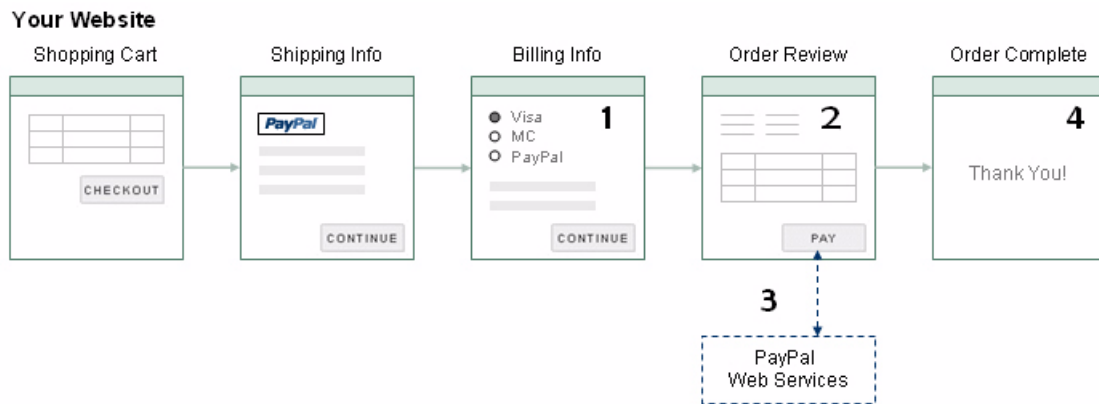


TABLE 4.1 Steps in Checkout with Direct Payment API

Step	Description
1	On your website, the customer chooses to pay with a credit card and enters the credit card number and other details.
2	The customer reviews the order.
3	When your customer clicks “Pay” to place the order, you call a PayPal API to request payment, and the payment transaction is initiated. NOTE: The customer does not see this step. PayPal does not send your customer a receipt for the payment.
4	You transfer your customer to your order confirmation page.

Technical Overview

At the point of payment in the checkout process, the “Pay” button on your website must send the `DoDirectPaymentRequest` to the PayPal Web Services API service, including required information you collected from the customer, such as the amount of the transaction, the

How Direct Payment API Works

Relationship to Authorization & Capture

buyer's credit card number, expiration date, browser IP address, and an element that specifies whether this transaction is a final sale (complete transaction amount including shipping, handling and tax) or an authorization for a final amount that you must capture later with Authorization & Capture. The Direct Payment API SOAP response includes a transaction identification number and other information.

PayPal is completely invisible to your customer, before, during, and after the purchase. PayPal does not send an email receipt to the customer, nor will the customer's credit card statement indicate that PayPal processed the payment.

For more details about the Direct Payment API, see the *Name-Value Pair Developer Guide and Reference* or *SOAP API Reference*.

IMPORTANT: Direct Payment API is not a standalone product. You are required to use Direct Payment and Express Checkout together as part of the Website Payments Pro solution. For information about these business rules, see [Chapter 2, "Button Placement, Page Designs, and Programming Flow."](#)

Relationship to Authorization & Capture

PayPal assumes that at the end of the checkout process, you will make a final sale and payment transaction via PayPal. If at point of sale you do not know the complete cost of the order—for example, if shipping, handling, and tax is not precisely known, or if you want to upsell—you can authorize a transaction that you capture later with Authorization & Capture.

For more information about Authorization & Capture, see [Chapter 6, "How Authorization & Capture Works."](#)

5

Recurring Payments

Overview

PayPal Recurring Payments allows you to bill a buyer for a fixed amount of money on a fixed schedule. The buyer signs up for recurring payments during checkout from your site. Consider the following examples:

- A buyer purchases a subscription to a magazine or newsletter from your site and agrees to pay a monthly fee.
- A buyer agrees to pay an Internet Service Provider a flat fee on a semi-annual basis to host a website.

These examples represent payment transactions that reoccur periodically and are for a fixed amount.

PayPal offers recurring payments as part of the following payment solutions:

- [Recurring Payments With Express Checkout](#)
- [Recurring Payments With Direct Payments](#)

IMPORTANT: To use the recurring payments features described in this chapter, you must set the version parameter to **50.0** in your NVP or SOAP API calls.

How Recurring Payments Work

When you create recurring payments for a buyer, you create a *recurring payments profile*. The profile contains information about the recurring payments, including details for an optional trial period and a regular payment period. Each of these subscription periods contains information about the payment frequency and payment amounts, including shipping and tax, if applicable.

After a profile is created, PayPal automatically queues payments based on the billing start date, billing frequency, and billing amount, until the profile expires or is canceled by the merchant. The buyer can also cancel the recurring payment profile for profiles creating using Express Checkout.

Note that for profiles created using Express Checkout, the queued payments are funded using the normal funding source hierarchy within the buyer's PayPal account.

After the recurring payments profile is created, you can view recurring payments details or cancel the recurring payments profile from your PayPal account. You can also access recurring payments reports from the PayPal **Business Overview** page.

Also, after creating a recurring payments profile, you can use the Recurring Payments API to do the following:

- Get information details about a recurring payments profile (see [“Getting Recurring Payments Profile Information” on page 53](#))
- Change the status of a recurring payment profile (see [“Recurring Payments Profile Status” on page 53](#))
- Update the details of the recurring payments profile (see [“Modifying a Recurring Payments Profile” on page 54](#))
- Bill the outstanding amount of the recurring payments profile. (see [“Billing the Outstanding Amount of a Profile” on page 55](#))

Limitations

The current release of the Recurring Payments API has the following limitations:

- A profile can have at most one optional trial period and a single regular payment period.
- The profile start date may not be earlier than the profile creation date.

Recurring payments with Express Checkout also has the following limitations:

- At most ten recurring payments profiles can be created during checkout.
- You can only increase the profile amount by 20% in each 180-day interval after the profile is created.

Recurring Payments Terms

[Table 5.1](#) lists commonly used terms related to recurring payments.

TABLE 5.1 Recurring Payments Terms

Term	Definition
Recurring payments profile	Your record of a recurring transaction for a single customer. The profile includes all information required to automatically bill the buyer a fixed amount of money at a fixed interval.
Billing cycle	One payment is made per billing cycle. Each billing cycle is made up of two components. <ul style="list-style-type: none">• The billing period specifies the unit to be used to calculate the billing cycle (such as days or months).• The billing frequency specifies the number of billing periods that make up the billing cycle. For example, if the billing period is Month and the billing frequency is 2, the billing cycle will be two months. If the billing period is Week and the billing frequency is 6, the payments will be scheduled every 6 weeks.

TABLE 5.1 Recurring Payments Terms

Term	Definition
Regular payment period	The main subscription period for this profile, which defines a payment amount for each billing cycle. The regular payment period begins after the trial period, if a trial period is specified for the profile.
Trial period	An optional subscription period before the regular payment period begins. A trial period may not have the same billing cycles and payment amounts as the regular payment period.
Payment amount	The amount to be paid by the buyer for each billing cycle.
Outstanding balance	If a payment fails for any reason, that amount is added to the profile's outstanding balance.
Profile ID	An alphanumeric string (generated by PayPal) that uniquely identifies a recurring profile.

Recurring Payments With Express Checkout

Overview

During the Express Checkout flow, you can create one or more recurring payments. You can also mix recurring payments with other purchases in the same Express Checkout flow.

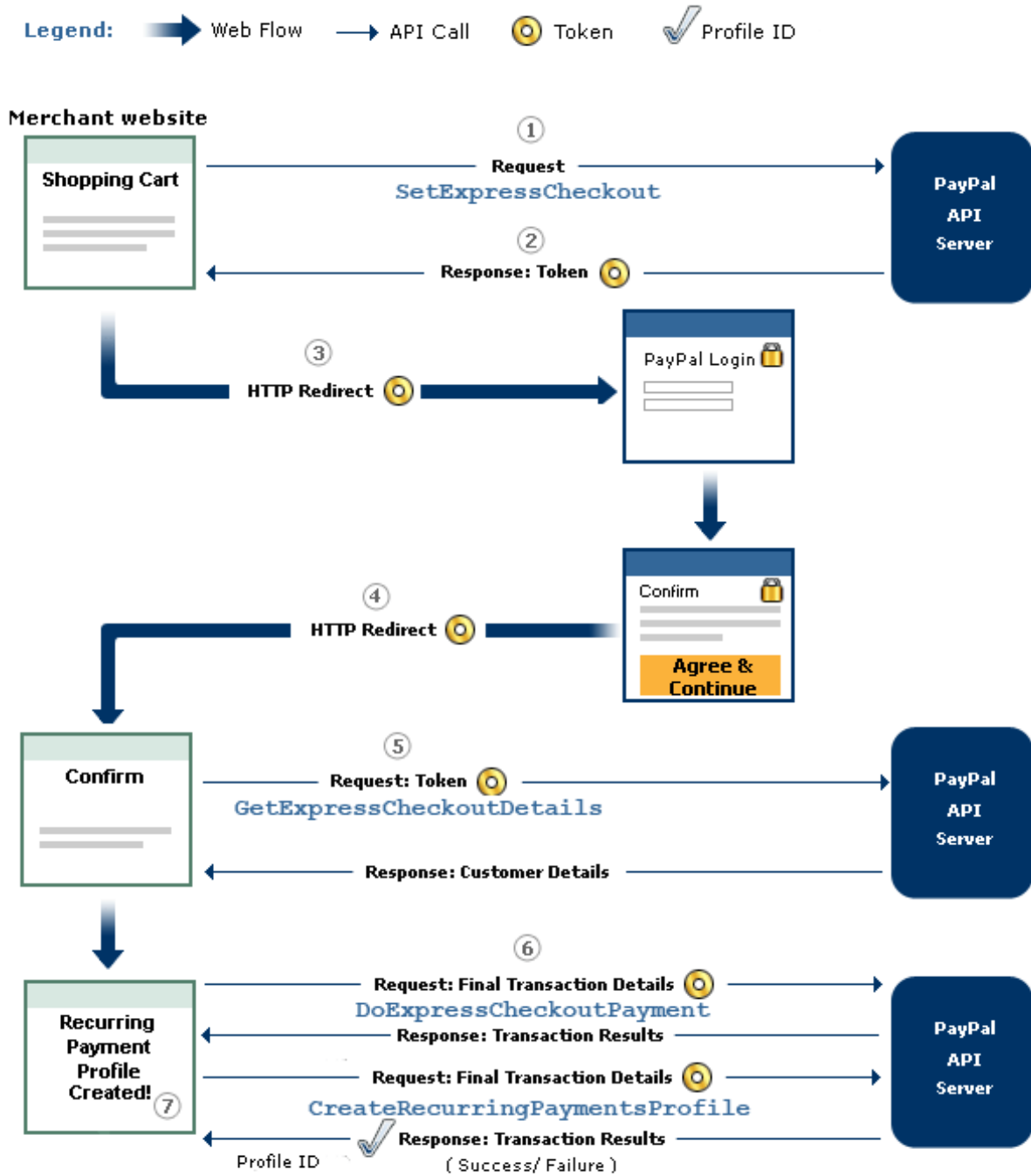
Creating Recurring Payments

The following diagram illustrates the typical processing flow to create recurring payments during checkout. The numbered steps in the figure are detailed in [Table 5.2](#).

Recurring Payments

Recurring Payments With Express Checkout

FIGURE 5.1 Recurring Payments with Express Checkout API Flow



Recurring Payments Processing Flow

Table 5.2 details the steps shown in Figure 5.1, “Recurring Payments with Express Checkout API Flow.”

TABLE 5.2 Recurring Payments Processing Flow

Step	Merchant...	PayPal...
1	Calls <code>SetExpressCheckout</code> with one or more billing agreement details in the request	
2		Returns a token, which identifies the transaction, to the merchant.
3	Redirects buyer’s browser to: <code>https://www.paypal.com/cgi-bin/webscr?cmd=_express_checkout&token=<token returned by SetExpressCheckout></code>	
		Displays login page. Allows user to select payment options and shipping address.
4		Redirects buyer’s browser to <code>returnURL</code> passed to <code>SetExpressCheckout</code> if buyer agrees to payment description.
5	Calls <code>GetExpressCheckoutDetails</code> to get buyer information (optional).	
		Returns <code>GetExpressCheckoutDetails</code> response.
	Displays merchant review page for buyer.	
6	Calls <code>DoExpressCheckoutPayment</code> if the order includes one-time purchases as well as a recurring payment. Otherwise, skip this step.	
		Returns <code>DoExpressCheckoutPayment</code> response
	Calls <code>CreateRecurringPaymentsProfile</code> one time for each recurring payment item included in the order.	
		Returns <code>ProfileID</code> in <code>CreateRecurringPaymentsProfile</code> response for each profile successfully created.
7	Displays successful transaction page.	

Initiating the Processing Flow With SetExpressCheckout

As in the Express Checkout flow, the SetExpressCheckout request notifies PayPal that you are initiating an order that can be either a one-time purchase, up to ten recurring payments, or a mixture of a one-time purchase and recurring payments.

NOTE: You can also initiate the processing flow using SetCustomerBillingAgreement for orders that contain only a single recurring payment.

To include one or more recurring payments in the SetExpressCheckout request, you must set the following fields:

TABLE 5.3 SetExpressCheckout Fields for Recurring Payments

NVP	SOAP	
L_BILLINGTYPE <i>n</i>	BillingAgreementDetails. BillingType	Type of billing agreement. For recurring payments, this field must be RecurringPayments.
L_BILLINGAGREEMENT DESCRIPTION <i>n</i>	BillingAgreementDetails. Description	Description of goods or services associated with the billing agreement.

NOTE: You must include these same values as part of the CreateRecurringPaymentsProfile request.

The SetExpressCheckout response provides a token that uniquely identifies the transaction for subsequent redirects and API calls.

Redirecting the Buyer's Browser to PayPal

After you receive a successful response from SetExpressCheckout, add the TOKEN from the SetExpressCheckout response as a name/value pair to the following URL, and redirect your buyer's browser to it:

```
https://www.paypal.com/cgi-bin/webscr?cmd=_express_checkout&token=<value_from_SetExpressCheckoutResponse>
```

For redirecting the buyer's browser to the PayPal login page, PayPal recommends that you use the HTTPS response 302 "Object Moved" with the URL above as the value of the Location header in the HTTPS response. Ensure that you use an SSL-enabled server to prevent browser warnings about a mix of secure and insecure graphics.

Getting Buyer Details Using GetExpressCheckoutDetails

The GetExpressCheckoutDetails method returns information about the buyer, including name and email address stored on PayPal. You can optionally call this API after PayPal redirects the buyer's browser to the ReturnURL you specified in the SetExpressCheckout request.

The GetExpressCheckoutDetails request has one required parameter, TOKEN, which is the value returned in the SetExpressCheckout response.

Creating the Profiles With CreateRecurringPaymentsProfile

After your buyer has agreed to the recurring payments billing agreement on your confirmation page, you must call `CreateRecurringPaymentsProfile` to create the profile. If you are creating multiple recurring payments profiles, you must call `CreateRecurringPaymentsProfile` once for each profile to be created.

If the transaction includes a mixture of a one-time purchase and recurring payments profiles, call `DoExpressCheckoutPayment` to complete the one-time purchase transaction, and then call `CreateRecurringPaymentsProfile` for each recurring payment profile to be created.

IMPORTANT: The recurring payments profile is not created until you receive a success response from the `CreateRecurringPaymentsProfile` call.

The `CreateRecurringPaymentsProfile` response contains a Profile ID, which is an encoded string that uniquely identifies the recurring payments profile.

For more options when creating a recurring payments profile, see [“Options for Creating a Recurring Payments Profile” on page 50.](#)

Recurring Payments With Direct Payments

For recurring payments with direct payments, you must collect on your website all necessary information from your buyer, including billing amount and buyer’s credit card information.

After you have collected the information, call the `CreateRecurringPaymentsProfile` API for each profile to be created. The `CreateRecurringPaymentsProfile` request must contain all required credit card information and must not contain a value for the `TOKEN` field.

[Table 5.4](#) lists the fields that are required in the `CreateRecurringPaymentsProfile` request for recurring payments using direct payments.

TABLE 5.4 Required Fields for CreateRecurringPaymentsProfile With Direct Payments

NVP	SOAP
CREDITCARDTYPE	CreditCardDetails.CreditCardType
ACCT	CreditCardDetails.CreditCardNumber
EXPDATE	CreditCardDetails.ExpMonth and CreditCardDetails.ExpYear
FIRSTNAME	CreditCardDetails.CardOwner.FirstName
LASTNAME	CreditCardDetails.CardOwner.LastName
PROFILESTARTDATE	RecurringPaymentProfileDetails.BillingStartDate
BILLINGPERIOD	ScheduleDetails.PaymentPeriod.BillingPeriod
BILLINGFREQUENCY	ScheduleDetails.PaymentPeriod.BillingFrequency

TABLE 5.4 Required Fields for CreateRecurringPaymentsProfile With Direct Payments

NVP	SOAP
AMT	ScheduleDetails.PaymentsPeriod.Amount

The CreateRecurringPaymentsProfile response contains a Profile ID, which is an encoded string that uniquely identifies the recurring payments profile.

For more options when creating a recurring payments profile, see [“Options for Creating a Recurring Payments Profile”](#) on page 50.

As with all direct payments, PayPal is completely invisible to your buyer before, during, and after the purchase. PayPal does not send an email receipt to the buyer, nor will the buyer’s credit card statement indicate that PayPal processed the payment.

Options for Creating a Recurring Payments Profile

This section describes many of the different options that you have when you create a recurring payments profile.

- [Specifying the Regular Payment Period](#)
- [Including an Optional Trial Period](#)
- [Specifying an Initial Payment](#)
- [Other Profile Options](#)

Specifying the Regular Payment Period

Each recurring payments profile has a regular payment period that defines the amount and frequency of the payment. [Table 5.5](#) lists the required fields for specifying the regular payment period.

TABLE 5.5 Specifying the Regular Payment Period

NVP	SOAP	Description
Required		
PROFILESTARTDATE	RecurringPaymentsProfileDetails.BillingStartDate	The date when billing for this profile begins. NOTE: The profile may take up to 24 hours for activation.

TABLE 5.5 Specifying the Regular Payment Period

NVP	SOAP	Description
BILLINGPERIOD	ScheduleDetails. PaymentPeriod. BillingPeriod	The unit of measure for the billing cycle. Must be one of: <ul style="list-style-type: none"> • Day • Week • SemiMonth • Month • Year
BILLINGFREQUENCY	ScheduleDetails. PaymentPeriod. BillingFrequency	Number of billing periods that make up one billing cycle. <p>NOTE: The combination of billing frequency and billing period must be less than or equal to one year.</p> <p>NOTE: If the billing period is SemiMonth., the billing frequency must be 1.</p>
AMT	ScheduleDetails. PaymentPeriod. Amount	Amount to bill for each billing cycle.

You can optionally include a value for TOTALBILLINGCYCLES (SOAP field ScheduleDetails.PaymentPeriod.TotalBillingCycles), which specifies the total number of billing cycles in the regular payment period. If no value is specified or if the value is 0, the payments continue until the profile is canceled or suspended. If the value is greater than 0, the regular payment period will continue for the specified number of billing cycles.

You can also specify an optional shipping amount or tax amount for the regular payment period.

Including an Optional Trial Period

You can optionally include a trial period in the profile by specifying the following fields in the CreateRecurringPaymentsProfile request. [Table 5.6](#) lists the required fields for creating an optional trial period.

TABLE 5.6 Specifying a trial period

NVP	SOAP
TRIALBILLINGPERIOD	ScheduleDetails.TrialPeriod.BillingPeriod
TRIALBILLINGFREQUENCY	ScheduleDetails.TrialPeriod.BillingFrequency
TRIALAMT	ScheduleDetails.TrialPeriod.Amount
TRIALTOTALBILLINGCYCLES	ScheduleDetails.TrialPeriod.TotalBillingCycles

Specifying an Initial Payment

You can optionally specify an initial non-recurring payment when the recurring payments profile is created by including the following fields in the `CreateRecurringPaymentsProfile` request:

TABLE 5.7 Specifying an Initial Payment

NVP	SOAP
<code>INITAMT</code>	<code>ScheduleDetails.ActivationDetails.InitialAmount</code>
<code>FAILEDINITAMTACTION</code>	<code>ScheduleDetails.ActivationDetails.FailedInitAmountAction</code>

By default, PayPal will not activate the profile if the initial payment amount fails. You can override this default behavior by setting the `FAILEDINITAMTACTION` field to `ContinueOnFailure`, which indicates that if the initial payment amount fails, PayPal should add the failed payment amount to the outstanding balance due on this recurring payment profile.

If this field is not set or is set to `CancelOnFailure`, PayPal will create the recurring payment profile, but will place it into a pending status until the initial payment is completed. If the initial payment clears, PayPal will notify you by IPN that the pending profile has been activated. If the payment fails, PayPal will notify you by IPN that the pending profile has been canceled.

The buyer will receive an email stating that the initial payment cleared or that the pending profile has been canceled if the profile was created using Express Checkout.

Other Profile Options

Maximum Number of Failed Payments

By including the `MAXFAILEDPAYMENTS` field in the `CreateRecurringPaymentsProfile` request, you set the number of failed payments allowed before the profile is automatically suspended.

Billing the Outstanding Amount

If a payment fails due to any reason, the amount that was to be billed (including shipping and tax, if applicable) is added to the profile's outstanding balance. Use the `AUTOBILLOUTAMT` field in the `CreateRecurringPaymentsProfile` request to specify whether or not the outstanding amount should be added to the payment amount for the next billing cycle.

Whether or not you choose to include the outstanding amount with the payment for the next billing cycle, you can also use the `BillOutstandingAmount` API to programmatically collect that amount at any time (see [“Billing the Outstanding Amount of a Profile” on page 55](#)).

Recurring Payments Profile Status

A recurring payments profile can have one of the following status values:

- `ActiveProfile`
- `PendingProfile`
- `SuspendedProfile`
- `CancelledProfile`
- `ExpiredProfile`

If the profile is successfully created, it has an `ActiveProfile` status. However, as described in [“Specifying an Initial Payment” on page 52](#), if a non-recurring initial payment fails and `FAILEDINITAMTACTION` is set to `CancelOnFailure` in the `CreateRecurringPaymentsProfile` request, the profile is created with a status of `PendingProfile` until the initial payment either completes successfully or fails.

A profile has a status of `ExpiredProfile` when both the total billing cycles for both the optional trial period and the regular payment period have been completed.

You can suspend or cancel a profile by using the `ManageRecurringPaymentsProfileStatus` API. You can also reactivate a suspended profile.

For recurring payments profiles created with Express Checkout, the buyer receives an email about the change in status of their recurring payment.

Getting Recurring Payments Profile Information

Use the `GetRecurringPaymentsProfileDetails` API to get information about a profile. Along with the information that you specified in the `CreateRecurringPaymentsProfile` request, `GetRecurringPaymentsProfileDetails` also returns the following summary information about the profile:

- Profile status
- Next scheduled billing date
- Number of billing cycles completed in the active subscription period
- Number of billing cycles remaining in the active subscription period
- Current outstanding balance
- Total number of failed billing cycles
- Date of the last successful payment received
- Amount of the last successful payment received

For recurring payments with direct payments, the buyer’s credit card information is also returned.

NOTE: Only the last 4 digits of the credit card account number are returned and CVV2 is never returned.

Modifying a Recurring Payments Profile

Use the `UpdateRecurringPaymentsProfile` API to modify a recurring payments profile.

NOTE: You can only modify active or suspended profiles.

You can only modify specific information about the profile, including the following:

- Subscriber name or address (see [“Updating Addresses” on page 54](#))
- Additional billing cycles to add
- Billing amount, tax amount, or shipping amount (see [“Updating the Billing Amount” on page 55](#))
- Past due or outstanding amount
- Whether to bill the outstanding amount with the next billing cycle
- Maximum number of failed payments allowed
- Buyer’s credit card information, except for the credit card number (see [“Updating Addresses” on page 54](#) for additional restrictions)
- Profile description and reference

NOTE: You cannot modify the billing frequency or billing period of a profile. You can only add additional billing cycles to the profile.

NOTE: For recurring payments with Express Checkout, certain updates, such as billing amount, are not allowed within 3 days of the scheduled billing date, and an error is returned.

For complete details, see the *Name-Value Pair Developer Guide and Reference* or the *SOAP API Reference*.

Updating Addresses

When you update the following addresses, you must enter all of address fields, not just those that are changing:

- Subscriber shipping address
- Credit card billing address (for recurring payments created using direct payments)

For example, if you want to update the subscriber’s street address, you must specify all of the address fields listed in the *Name-Value Pair Developer Guide and Reference* or *SOAP API Reference*, not just the field for the street address.

Updating the Billing Amount

For profiles created using Express Checkout, the total amount of a recurring payment can only be increased 20% in a fixed 180-day interval after the profile is created. The 20% maximum is based on the total amount of the profile at the beginning of the 180-day interval, including any shipping or tax amount.

NOTE: This restriction is only for recurring payments profiles created with Express Checkout. There is no restriction for profiles created using direct payments.

For example, if a profile is created on March 10 with a total amount of \$100, then during the 180-day interval from March 10 to September 6, you can increase the payment amount to a maximum of \$120 (120% of \$100).

Suppose that during the first 180-day interval, you increased the payment amount to \$110. Then during the next 180-day interval (starting on September 7 in this example), you can only increase the amount of the payment to a maximum of \$132 (120% of \$110).

Billing the Outstanding Amount of a Profile

Use the `BillOutstandingAmount` API to immediately bill the buyer for the current past due or outstanding amount for a recurring payments profile. To bill the outstanding amount:

- The profile status must be active or suspended.
- The profile must have a non-zero outstanding balance.
- The amount of the payment cannot exceed the outstanding amount for the profile.
- The `BillOutstandingAmount` call cannot be within 24 hours of a regularly scheduled payment for this profile.

NOTE: If another outstanding balance payment is already queued, an API error is returned.

You will be informed by IPN about the success or failure of the outstanding payment. For profiles created using Express Checkout, the buyer will receive an email notification of the payment.

PayPal Notifications

Merchants are notified of certain events through IPN. For recurring payments profiles created using Express Checkout, buyers are also notified of specific events by email. [Table 5.8](#) indicates when IPN or emails are generated.

NOTE: API transactions such as `ManangeRecurringPaymentsProfileStatus` do not trigger IPN notification because the success or failure of the call is immediately provided by the API response.

NOTE: No emails to buyers are ever sent for recurring payments profiles created using direct payments.

TABLE 5.8 PayPal Notifications

Event	IPN	Buyer Email
Profile successfully created	Yes	Yes
Profile creation failed	Yes	Yes
Profile canceled from paypal.com interface	Yes	Yes
Profile status changed using API	No	Yes
Profile update using API	No	Yes
Initial payment either succeeded or failed	Yes	Yes
Payment either succeeded or failed (during either trial period or regular payment period)	Yes	Yes
Outstanding payment either succeeded or failed	Yes	Yes

6

How Authorization & Capture Works

Authorization & Capture is a settlement solution that provides merchants increased flexibility in obtaining payments from their buyers. During a traditional sale at PayPal, the authorization and capture action is completed simultaneously. Authorization & Capture separates the authorization of payment from the capture of the authorized payment.

Authorization & Capture is for merchants who have a delayed order fulfillment process and who typically make a \$1 auth at checkout. It enables merchants to modify the original authorization amount due to order changes occurring after the initial order is placed (such as taxes, shipping, or item availability). This chapter discusses the authorization and capture process and provides steps to help you authorize, capture, reauthorize, and void funds.

There are two ways to use Authorization & Capture:

1. Use the Authorization & Capture Application Programming Interface (API), which is discussed here and detailed in the *PayPal Name-Value Pair Developer Guide and Reference* and the *PayPal SOAP API Reference*.
2. Create an order or authorization with Website Payments Standard HTML and capture or void the authorization on the PayPal website (<https://www.paypal.com/>). This topic is not discussed here. For more information about the Authorization & Capture and Website Payments Standard, see the *Website Payments Standard Integration Guide*.

Fundamental Authorization Process With the APIs

Authorization & Capture starts when your buyer authorizes a payment amount during checkout.

1. For example, you can use the PayPal Express Checkout API with the `PAYMENTACTION` element set to `Authorization` or `Order`.

Similarly, you can use the Direct Payment API with the `PAYMENTACTION` element set to `Authorization`. Currently, the Direct Payment API does not support order authorizations.

2. After your buyer completes checkout, you can then use the payment's transaction ID with Authorization & Capture APIs. You can:
 - Capture either a partial amount or the full authorization amount.
 - Authorize a higher amount, up to 115% of the originally authorized amount (not to exceed an increase of \$75 USD).
 - Void a previous authorization.

Honor Period and Authorization Period

When your buyer approves an authorization, the buyer's balance can be placed on hold for a 29-day period to ensure the availability of the authorization amount for capture. You can reauthorize a transaction only once, up to 115% of the originally authorized amount (not to exceed an increase of \$75 USD).

After a successful authorization (or reauthorization), PayPal will honor authorized funds for three days, but PayPal cannot ensure that 100% of the funds will be available. A day is defined as the start of the calendar day on which the authorization or reauthorization was made (from 12AM PST to 11:50PM PST).

You can settle without a reauthorization from day 4 to day 29 of the authorization period, but PayPal cannot ensure that 100% of the funds will be available after the three-day honor period.

If you attempt to capture funds outside the honor period, PayPal applies best efforts to capture funds. However, there is a possibility that funds will not be available at that time.

Buyer and seller accounts cannot be closed if there is a pending (unsettled) authorization.

Supported PayPal Payment Products

You can use Authorization & Capture with the PayPal products listed in [Table 6.1, "PayPal Products Supporting Authorization & Capture."](#)

By default, these products assume that a transaction is a final sale. You must explicitly specify that a transaction is a basic or order authorization.

NOTE: You must capture and void orders and order authorizations using the Authorization & Capture APIs. That is, you cannot process order authorizations on the PayPal website (<https://www.paypal.com>). The PayPal website supports processing only basic authorizations, not order authorizations.

TABLE 6.1 PayPal Products Supporting Authorization & Capture

Product	Typical Usage
Website Payments	PAYMENTACTION="authorization"
Buy Now	PAYMENTACTION="authorization"
Donations	PAYMENTACTION="authorization"
Shopping carts	PAYMENTACTION="authorization"

PayPal Products Not Supported. Authorization & Capture cannot be used with the following products:

- eBay checkout
- eCheck
- Gift Certificates and Coupons
- Subscriptions
- Instant Purchase

- Send Money
- Request Money
- Virtual Terminal
- Invoicing

Order Authorizations Scenarios

The following are common scenarios you will encounter when implementing order authorizations.

Simple Order

TABLE 6.2 Simple Order Scenario

Action	API Call	Running Balance
Your buyer orders an item from your website.		
You request an order from PayPal to authorize the payment, specifying the variable <code>paymentaction=order</code> . You receive a response that the order has been created for a payment amount of \$100.00.	DoExpressCheckoutPaymentRequest with: <code>PAYMENTACTION=order</code>	
You request authorization for \$100.00.	DoAuthorizationRequest	<\$100.00>
You capture funds in the amount of \$115.00, the maximum amount allowed. The order now has a “Complete” status.	DoCaptureRequest	\$15.00

Complex Order

TABLE 6.3 Complex Order Scenario

Action	API Call	Running Balance
Your buyer orders 2 items from your website.		

How Authorization & Capture Works

Order Authorizations Scenarios

TABLE 6.3 Complex Order Scenario

Action	API Call	Running Balance
You request an order from PayPal to authorize the payment, specifying the variable <code>paymentaction=order</code> . You receive a response that the order has been created for a payment amount of \$100.00.	DoExpressCheckoutPaymentRequest with: <code>PAYMENTACTION=order</code>	
You request authorization #1 for \$75.00.	DoAuthorizationRequest	<\$75.00>
Your buyer contacts you and upgrades to next-day shipping. You capture funds in the amount of \$80.00 on authorization #1 to accommodate for the additional shipping charges.	DoCaptureRequest	\$5.00
You request authorization #2 for \$25.00.	DoAuthorizationRequest	<\$25.00>
Your buyer contacts you and changes an item on the order. You void authorization #2.	DoVoid	\$0.00
You request authorization #3 for \$35.00 for the new item selection.	DoAuthorizationRequest	<\$35.00>
You capture authorization #3 for \$35.00. Because the maximum amount of funds that can be captured has been reached (115% of the original order amount), the order now has a “Complete” status.	DoCaptureRequest	\$0.00

Concurrent Authorizations

TABLE 6.4 Concurrent Authorizations Scenario

Action	API Call	Running Balance
Your buyer orders 3 pieces of equipment for \$300.00 from your website.		
You request an order from PayPal to authorize the payment, specifying the variable <code>paymentaction=order</code> . You receive a response that the order has been created for a payment amount of \$300.00.	DoExpressCheckoutPaymentRequest with: <code>PAYMENTACTION=order</code>	
You request authorization #1 on day 1 for \$100.00 for the keyboard. You receive a response that the authorization has been created for a payment amount of \$100.00.	DoAuthorizationRequest	<\$100.00>

TABLE 6.4 Concurrent Authorizations Scenario

Action	API Call	Running Balance
You capture authorization #1 for \$100.00. You ship the keyboard.	DoCaptureRequest	\$0.00
You request authorization #2 on day 2 for \$200.00 for the second component. You receive a response that the authorization has been created for a payment amount of \$200.00.	DoAuthorizationRequest	<\$200.00>
On day 3, you capture authorization #2 for \$200.00.	DoCaptureRequest	\$0.00
You ship the second component.		

Total Capture Hits Relative Tolerance With Open Authorizations

TABLE 6.5 Total Capture Hits Relative Tolerance with Open Authorizations Scenario

Action	API Call	Running Balance
Your buyer orders an item from your website for \$1000.00.		
You request an order from PayPal to authorize the payment, specifying the variable paymentaction=order. You receive a response that the order has been created for a payment amount of \$1000.00.	DoExpressCheckoutPaymentRequest with: PAYMENTACTION=order	
Your buyer requests overnight shipping. You request authorization #1 for \$1100.00 (110% relative tolerance).	DoAuthorizationRequest	<\$1100.00>
You receive a response that the authorization has been created for a payment amount of \$1100.00. You ship the in-stock item by overnight shipping.		
You capture authorization #1 for \$1100.00.	DoCaptureRequest	\$0.00
Your buyer contacts you and adds another item to the order. You request authorization #2 for \$60.00, increasing the order total to \$1160.00, which exceeds the 115% tolerance limit of the original order.	DoAuthorizationRequest	<\$60.00>

How Authorization & Capture Works

Order Authorizations Scenarios

TABLE 6.5 Total Capture Hits Relative Tolerance with Open Authorizations Scenario

Action	API Call	Running Balance
You receive a response that the request for authorization #2 has been declined.		\$0.00

Void Authorizations

TABLE 6.6 Void Authorizations Scenario

Action	API Call	Running Balance
Your buyer orders 2 items from your website.		
You request an order from PayPal to authorize the payment, specifying the variable <code>paymentaction=order</code> . You receive a response that the order has been created for a payment amount of \$350.00.	DoExpressCheckoutPaymentRequest with: <code>PAYMENTACTION=order</code>	
You request authorization #1 on day 1 for \$100.00. You receive a response that the authorization has been created for a payment amount of \$100.00.	DoAuthorizationRequest	<\$100.00>
You ship the item. You capture authorization #1 for \$100.00.	DoCaptureRequest	\$0.00
You request authorization #2 on day 2 for \$200.00. You receive a response that the authorization has been created for a payment amount of \$200.00.	DoAuthorizationRequest	<\$200.00>
The buyer contacts you and cancels the remaining item.		
You void authorization #2.	DoVoid	\$0.00

Partial Capture

TABLE 6.7 Partial Capture Scenario

Action	API Call	Running Balance
Your buyer orders an item from your website.		

TABLE 6.7 Partial Capture Scenario

Action	API Call	Running Balance
You request an order from PayPal to authorize the payment, specifying the variable <code>paymentaction=order</code> . You receive a response that the order has been created for a payment amount of \$100.00.	DoExpressCheckoutPaymentRequest with: <code>PAYMENTACTION=order</code>	
.You request authorization #1 on day 1 for \$100.00. You receive a response that the authorization has been created for a payment amount of \$100.00.	DoAuthorizationRequest	<\$100.00>
You ship the item. You capture authorization #1 for \$100.00.	DoCaptureRequest	\$0.00
With <code>CompleteType</code> set to <code>NotComplete</code> on the DoCapture API, you capture funds in the amount of \$50.00.	DoCaptureRequest	\$50.00

Complete Capture

TABLE 6.8 Complete Capture Scenario

Action	API Call	Running Balance
Your buyer orders an item from your website.		
You request an order from PayPal to authorize the payment, specifying the variable <code>paymentaction=order</code> . You receive a response that the order has been created for a payment amount of \$100.00.	DoExpressCheckoutPaymentRequest with: <code>PAYMENTACTION=order</code>	
You request authorization #1 on day 1 for \$100.00. You receive a response that the authorization has been created for a payment amount of \$100.00.	DoAuthorizationRequest	<\$100.00>
You capture authorization #1 for \$100.00. You ship the item.	DoCaptureRequest	\$0.00

NOTE: The default for the DoCapture API is a Complete capture, not a Partial capture.

Optimal Buyer Experience

This section details the best practices you should follow in using Authorization & Capture to ensure the best buying experience for your customers and getting the most from Authorization & Capture.

Capturing Funds on Basic Authorizations

PayPal recommends that you capture funds within the honor period of three days because PayPal will honor the funds for a 3-day period after the basic authorization. If you attempt to capture funds after the three-day period and the authorization fails, your request to capture funds may be declined.

After day 4 of the authorization period, you can initiate a reauthorization, which will start a new three-day honor period. However, it will not extend the original authorization period past 29 days. For example, if you successfully complete a reauthorization on day 29 of the authorization period, funds will only be honored until the end of the 29th day, and a new three-day honor period will start but not extend beyond day 29.

You should capture funds within 24 hours after you ship your buyer's order.

Buyer Approval for Basic Authorizations

A buyer-initiated authorization allows you to capture funds from the buyer's account up to 115% of the originally authorized amount (not to exceed an increase of \$75 USD) and up to \$10,000 USD.

IMPORTANT: If you want to update any details of the purchase that change the original authorization amount, PayPal requires that you obtain consent from the buyer at the time of purchase or at the time of capture.

Voiding Basic Authorizations

You should void an authorization if the authorization or reauthorization will not be used.

Voiding the authorization unlocks the temporary hold placed on your buyer's funding sources.





How Authorization & Capture Works

Optimal Buyer Experience

7

Integrating giropay with Express Checkout

This chapter describes the processing flow and required changes to your integration for Express Checkout transactions using giropay, a common German funding source. This chapter contains the following sections:

- [“Processing Page Flow” on page 67](#) illustrates the high-level processing flow for giropay transactions.
- [“Giropay Integration” on page 68](#) describes the API and other programming changes you must make to add giropay as a funding source.

Processing Page Flow

When your customer selects giropay as a funding source during the Express Checkout flow, you redirect the customer to a static PayPal URL after your order review page. PayPal then redirects the customer to the giropay website to push the funds to the merchant. After the giropay payment is successfully completed, the transaction is confirmed.

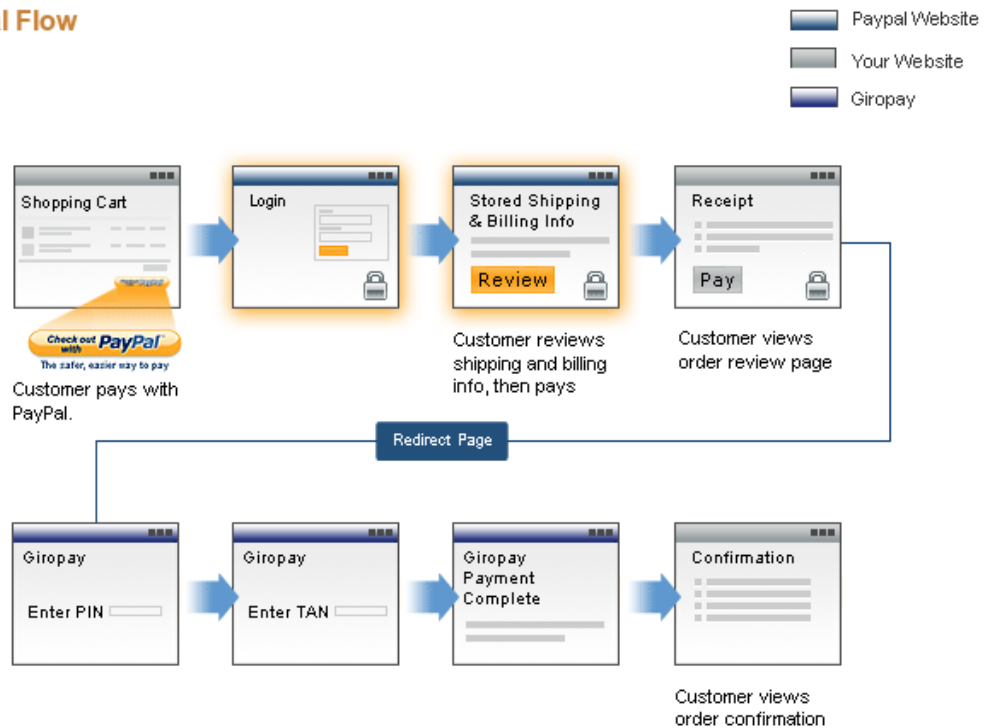
If the giropay payment fails or is cancelled by the customer, PayPal provides the necessary details for an EFT so that the customer can complete the transaction by pushing funds from his/her bank account. If your PayPal account profile blocks non-instant payments, the transactions is cancelled.

Giropay Payment Page Flow

[Figure 7.1](#) illustrates the flow of a successful giropay payment.

FIGURE 7.1 Page flow for a successful giropay payment

PayPal Flow



Cancelled or Unsuccessful Giropay Payment Page Flow

If the giropay payment fails for any reason, such as insufficient funds or the customer cancels, PayPal provides details to the customer to do a bank transfer from their bank account. This transaction will remain pending until PayPal receives the funds, at which time the transaction will be complete.

If you have disabled non-instant funding transactions for your PayPal account, the transaction is cancelled and PayPal redirects the customer to your *Order Cancel* page.

After the bank transfer flow is completed, the transaction is pending until the customer pushes the funds to PayPal.

If the customer cancels during the PayPal payment via bank transfer flow, your *Order Cancel* page is displayed.

Giropay Integration

This section details the programming and API changes necessary to implement the processing flow detailed in “[Processing Page Flow](#)” on page 67.

- [Initiate the Flow with SetExpressCheckout](#)

- Redirect the Customer to PayPal
- Complete the Transaction
- Receive Transaction Status Notification

Initiate the Flow with SetExpressCheckout

To support giropay payments, you pass the following three URLs as part of the `SetExpressCheckout` request. These URLs tell PayPal where to redirect the customer based on the success or failure of each type of payment transaction. See the *PayPal Name-Value Pair Developer Guide and Reference* for more information.

TABLE 7.1 SetExpressCheckout fields for giropay

NVP	Description
GIROPAYSUCCESSURL	The URL on the merchant site to redirect to after a successful giropay payment.
GIROPAYCANCELURL	The URL on the merchant site to redirect to after a giropay or bank transfer payment is cancelled or fails.
BANKTXNPENDINGURL	The URL on the merchant site to transfer to after a bank transfer payment.

Redirect the Customer to PayPal

After selecting a funding source on PayPal, the customer is redirected back to your website, as in the regular Express Checkout flow. There is one additional field, `REDIRECTREQUIRED`, returned in the response from both `GetExpressCheckoutDetails` and `DoExpressCheckoutPayment`.

If the value of this field is `true`, you redirect the customer from your *Order Review* page to <https://www.paypal.com/webscr?cmd=complete-express-checkout>. PayPal then redirects the customer from this redirect page to the necessary page for the selected funding source.

TABLE 7.2 GetExpressCheckoutDetails and DoExpressCheckoutPayment response

NVP	Description
REDIRECTREQUIRED	Flag to indicate whether you need to redirect the customer to back to PayPal

The `GetExpressCheckoutDetails` response contains the `REDIRECTREQUIRED` field, which lets you know if you need to redirect the user after your *Order Review* page. You can use this value to inform the customer on your *Order Review* page that they will be sent to the giropay website to complete the order.

Complete the Transaction

Corresponding to the three fields passed to SetExpressCheckout (see [Table 7.1](#)), you must add the following three additional pages to your website:

TABLE 7.3 Additional pages required for giropay integration

Page	Description
Order Completion	The page to redirect the customer to after a successful giropay payment.
Order Cancellation	The page to redirect the customer to after a giropay or bank transfer payment is cancelled or fails.
Order Pending	The page to redirect the customer to after a bank transfer payment.

Receive Transaction Status Notification

After PayPal redirects the customer to giropay, you receive transaction status information in the following ways:

- IPN Notification
- Email (only for successful giropay or bank transfer transactions)
- PayPal Account Overview
- PayPal reports