PayPal™

SOAP API Developer Reference

For Professional Use Only
Currently only available in English.

A usage Professional Uniquement
Disponible en Anglais uniquement pour l'instant.

PayPal SOAP API Developer Reference

Document Number: 100002.en_US-20070411

Contents

April 2007

Preface

This Document

This document describes the PayPal SOAP Application Programming Interface (API) and service.

Intended Audience

This document is written for programmers familiar with application programming standards such as the Simple Object Access Protocol (SOAP), the Web Services Description Language (WSDL), and XML Schema Definition (XSD) language.

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

Typeface	How Used
San-serif oblique	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_X is the name of the variable for a multi-item shopping cart. amount_3 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.

Documentation Problems

If you discover any errors in or have any problems with this documentation, please email us by following the instructions below. Describe the error or problem as completely as possible and give us the document title, the date of the document (located at the foot of every page), and the page number or page range.

To contact Developer Technical Support about documentation problems:

1. Log in to your account at https://developer.paypal.com/ by entering your email address and password in the Member Log In box

2. Click Help Center at the bottom of the box on the right side of the page.

3. Click Email PayPal Technical Support.

4. Complete the form.

Revision History

Revision history for *PayPal SOAP API Developer Reference*.

TABLE P.1 Revision History

Date	Description
April 2007	Added recurring payments APIs: SetCustomerBillingAgreement, GetBillingAgreementCustomerDetails, and CreateRecurringPaymentsProfile.

***TABLE* P.1 Revision History**

Date	Description
March 2007	Minor bug fixes including adding Switch/Solo codes to AVS Response Codes and CVV2 Response Codes in Direct Payment API chapter.
February 2007	Minor bug fixes.
December 2006	Minor bug fixes.
October 2006	Book renamed SOAP API Developer Reference. Former books for the SOAP SDKs now included in this one volume.
June 2006	CardNumber field added to TransactionSearch API. Significantly improved error messages for Direct Payment API. Minor change to one Mass Pay API error message.
March 2006	Updated for new API credential: API signatures. New SOAP service endpoint for signatures. Miscellaneous minor corrections throughout.
January 2006	Additional error messages for Authorization & Capture APIs and Express Checkout APIs.
December 2005	Removed erroneous description that stated that the SetExpressCheckoutRequest field `cpp-header-image` must be URL-encoded.

1   PayPal SOAP API Architecture

The PayPal SOAP API provides programmatic access to PayPal features and services. Developers can build custom applications, tools, and services that correspond to the same services and tools available through the main PayPal website, `https://www.paypal.com/`. Typical applications include searching for transactions, paying en masse, and making refunds.The API is based on open standards known collectively as "Web Services," which include the Simple Object Access Protocol (SOAP), Web Services Definition Language (WSDL), and the XML Schema Definition language (XSD). These standards are supported by a wide range of development tools on a variety of platforms.

Services Architecture

Like many web services, PayPal SOAP is a combination of client-side and server-side schemas, hardware and software servers, and core services.

FIGURE 1.1 PayPal SOAP High-level Diagram
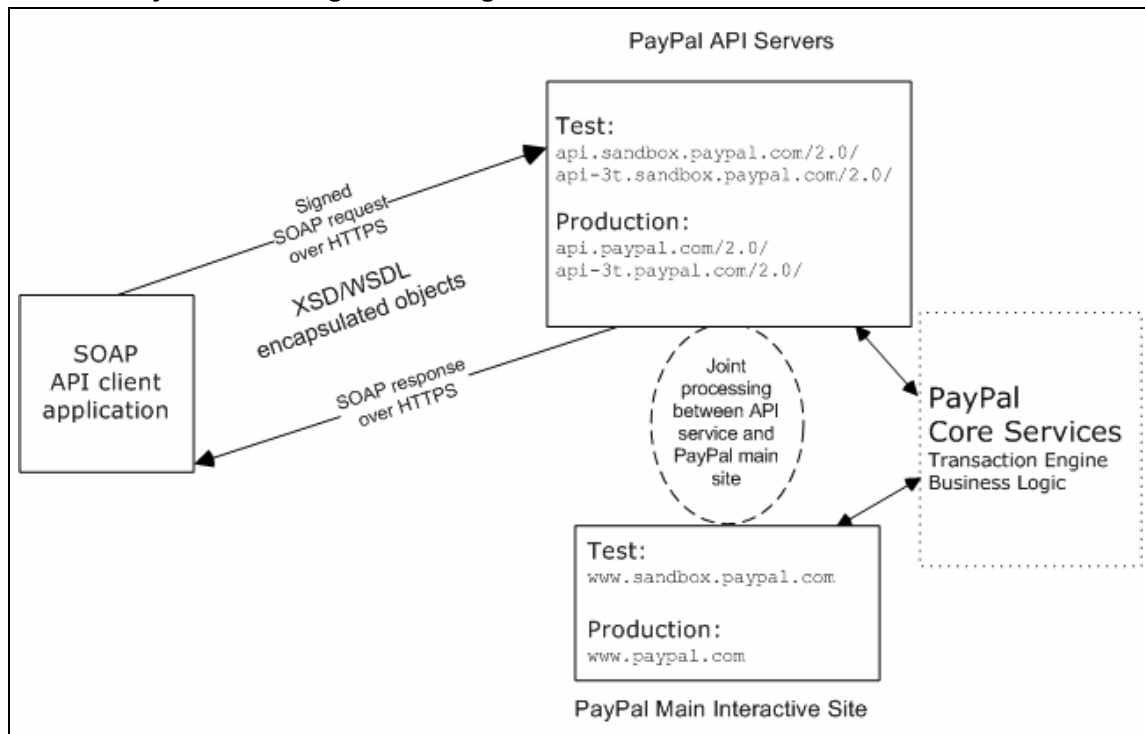
In an object-oriented processing model, the interface to SOAP requests/responses is an object in your application's native programming language. Your third-party SOAP client generates

business-object interfaces and network stubs from PayPal-provided WSDL and XSD files that specify the PayPal SOAP message structure, its contents, and the PayPal API service bindings. A business application works with data in the form of object properties to send and receive data by calling object methods. The SOAP client handles the details of building the SOAP request, sending it to the PayPal service, and converting the response back to an object.

PayPal WSDL/XSD Schema Definitions

The PayPal Web Services schema and its underlying eBay Business Language (eBL) base and core components are required for developing applications with the PayPal Web Services API. The following are the locations of the WSDL and XSD files.

TABLE 1.1 Location of PayPal WSDL and XSD Files

Development and Test with the PayPal Sandbox API Service	
PayPal Schema	https://www.sandbox.paypal.com/wsdl/PayPalSvc.wsdl
eBL Base Components and Component Types	https://www.sandbox.paypal.com/wsdl/eBLBaseComponents.xsd https://www.sandbox.paypal.com/wsdl/CoreComponentTypes.xsd
Production with Live PayPal Web Services API Service	
PayPal Schema	https://www.paypal.com/wsdl/PayPalSvc.wsdl
eBL Base Components and Component Types	http://www.paypal.com/wsdl/eBLBaseComponents.xsd http://www.paypal.com/wsdl/CoreComponentTypes.xsd

API Concepts and Terminology

Here are some basic concepts and terminology relating to PayPal's API service and security authentication.

TABLE 1.2 Basic PayPal API Set-up Concepts and Terminology

Term	Definition
API Calls	PayPal Application Programming Interface services, by which companies can make payments, search transactions, refund payments, view transaction information, and other business functions.
API Certificate	Mututally exclusive with API Signature. A PayPal-generated unique digital certificate file that you download from the PayPal website and use on the client computer to encrypt the HTTPS requests of your API calls to PayPal's API server. An API certificate is suitable if you have complete control over your own web server.

TABLE 1.2 Basic PayPal API Set-up Concepts and Terminology

Term	Definition
API Signature	Mututally exclusive with API Signature. A PayPal-generated unique digital signature (a line of text, or hash) that you copy from PayPal's website and include in your API calls. An alternative to API Certificate security. Your digital signature, your API username, and your API password all together are called three-token authentication, because you include each of them as a programatic token in your API calls. An API signature is suitable for use with Microsoft Windows web servers or other shared web server configurations, such as those used by web hosting services.
API Username and Password	A PayPal-generated identifying account name and password that you use specifically for making API calls. You include your API username and password with every API call. The API username and password are different from your PayPal login username (email address) and password.
Subject authorization	An indicator in an API call of the account for whom the call is being made. This is the programmatic aspect of third-party authorization. The value of the Subject field is the third-party's Paypal email address.
First-Party Access	A company makes API calls itself from its own server to PayPal's server. The company has its own API certificate or API signature, username, and password. Example: A staff programmer for a merchant's company obtains a PayPal-issued API certificate file and makes API calls for the company from the company's own web server.
Third-Party Access	Another person or company makes API calls on the your behalf. You grant the third-party your permission to make API calls for you. Example: A web hosting service has its own API certificate, API username, and API password. Its customers, who are merchants that use PayPal, give the hosting service their permission to make API calls on their behalf. The hosting service includes a merchant's PayPal email address in the "Subject" field of an API call.

Security

The PayPal SOAP API service is protected to ensure that only authorized PayPal members use it. There are four levels of security:

1. A required API username (Username field) and API password (Password field)

2. A third required authentication mechanism, which is either one of the following:

   – Client-side request signing via a PayPal-issues API Certificate
   – Request authentication via an an API Signature included in the request (Signature field)

3. An optional third-party authorization to make the API call on some other account's behalf (the optional Subject field).

4. Secure Sockets Layer (SSL) data transport

A failure of authenticated security at any one of these levels denies access to the PayPal SOAP API service.

SOAP RequesterCredentials: Username, Password, Signature, and Subject

For the security of your business, PayPal must verify that merchants or third-party developers are permitted to initiate a transaction before they make one. PayPal authenticates each request. If the request cannot be authenticated, a SOAP security fault is returned.

In the SOAP request header, your SOAP client must set the `Username`, `Password` elements to pass an API username/password combination. In addition, you can set the `Signature` or `Subject` elements to specify your API signature string and an optional third-party account email address for authentication. The following is a partial example of the RequesterCredentials elements required for all SOAP requests. For a correlation of these elements to the generic structure of an entire SOAP request, see "SOAP Request Envelope" on page 17.

```
<SOAP-ENV:Header>
     <RequesterCredentials xmlns="urn:ebay:api:PayPalAPI"
xsi:type="ebl:CustomSecurityHeaderType">
        <Credentials xmlns="urn:ebay:apis:eBLBaseComponents"
xsi:type="ebl:UserIdPasswordType">
            <Username>api_username</Username>
            <Password>api_password</Password>
            <Signature>api_signature</Signature>
            <Subject>authorizing_account_emailaddress</Subject>
        </Credentials>
     </RequesterCredentials>
</SOAP-ENV:Header>
```

where:

*TABLE **1.3** RequesterCredentials Authentication Elements in SOAP Header*

Element	Value	Description
`<Username>`	api_username	Your API username, which is auto-generated by PayPal when you apply for a digital certificate to use the PayPal SOAP API. You can see this value on https://www.paypal.com/ in your **Profile** under **API Access** > **API Certificate Information**. **NOTE:** The *PayPal Sandbox User Guide* contains details about how you can obtain digital certificates for use with the PayPal SOAP API.
`<Password>`	api_password	Your API password, which you specify when you apply for a digital certificate to use the PayPal SOAP API.
`<Signature>`	api_signature	Your API signature, if you use one instead of an API Certificate.

TABLE **1.3 RequesterCredentials Authentication Elements in SOAP Header**

Element	Value	Description
`<Subject>`	`authorizing_ account_ emailaddress`	The email address of a third-party for whom you are sending requests to the PayPal SOAP API. Your API username must have been granted permission by this third-party to make any particular PayPal API request.

SOAP Service Endpoints

Depending on your chosen authentication mechanism, your SOAP requests must be processed by different service endpoints.

TABLE **1.4 SOAP Service Endpoints**

Authentication Mechanism	Live Production Endpoint	Test (Sandbox) Endpoint
API Signature	https://api-3t.paypal.com/2.0/	https://api-3t.sandbox.paypal.com/2.0/
API Certificate	https://api.paypal.com/2.0/	https://api.sandbox.paypal.com/2.0/

NOTE: For the correspondence between API calls and their service endpoints, see Table 1.8, "API Overview by Business Function," on page 24.

SOAP Implementation

This section contains information about the PayPal SOAP implementation.

SOAP Message Style: doc-literal

PayPal uses `doc-literal` SOAP messaging, not `rpc-encoding`. With `doc-literal`, a single service interface call passes an XML document in the request to the PayPal API server, which responds with an XML document instance.

SOAP Request Envelope

The following diagram illustrates the contents of a PayPal SOAP request envelope.

All PayPal APIs are based on two core structures: `AbstractRequestType` and `AbstractResponseType`.

FIGURE 1.2 Diagram of SOAP Request Envelope
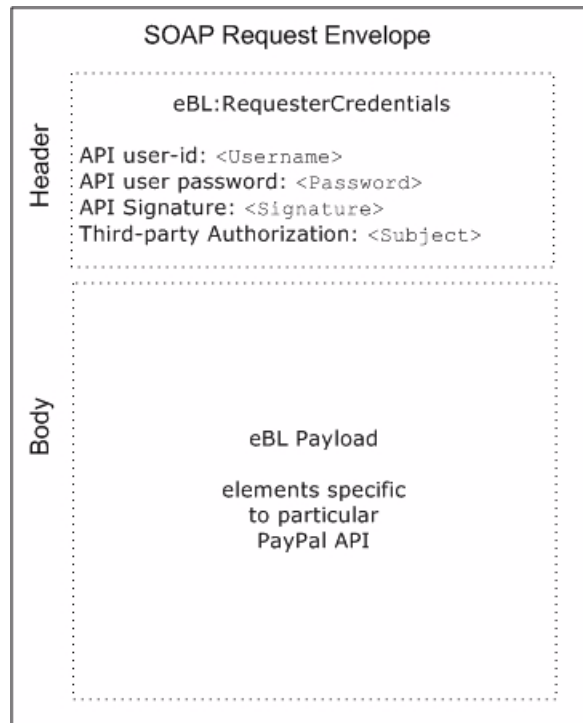
Request Structure

The following is an annotated description of the SOAP request structure required by the PayPal SOAP API.

General Structure of PayPal API SOAP Request

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
><SOAP-ENV:Header>
    <RequesterCredentials xmlns="urn:ebay:api:PayPalAPI">
        <Credentials xmlns="urn:ebay:apis:eBLBaseComponents">
            <Username>api_username</Username>
            <Password>api_password</Password>
            <Signature/>
            <Subject/>
        </Credentials>
    </RequesterCredentials>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
    <specific_api_name_Req xmlns="urn:ebay:api:PayPalAPI">
```

```
            <specific_api_name_Request>
               <Version xmlns=urn:ebay:apis:eBLBaseComponents">service_version
               </Version>
               <required_or_optional_fields xsi:type="some_type_here">data
               </required_or_optional_fields>
            </specific_api_name_Request>
         </specific_api_name_Req>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

*TABLE **1.5** Annotation of Generic SOAP Request*

Lines	Comment
12, 13	The `<Username>` and `<Password>` fields are part of the PayPal SOAP API `<RequesterCredentials>` security authentication mechanism you must construct for every SOAP request header. For details, see "SOAP RequesterCredentials: Username, Password, Signature, and Subject" on page 16.
14	The `<Signature>` element should include your API signature string if that is the kind of API credential you are using. For more details, see Table 1.3, "RequesterCredentials Authentication Elements in SOAP Header," on page 16.
15	The `<Subject>` element can specify a third-party PayPal account by whom you are authorized to make this request. For more details, see Table 1.3, "RequesterCredentials Authentication Elements in SOAP Header," on page 16.
19 through 27	The SOAP request for every PayPal API follows this element naming pattern. The API's specific name is appended with `Req`, and in this element the specific_api_name_Request is nested. Each specific_api_name_Request has a corresponding specific_api_name_RequestType.
22	The number of the PayPal SOAP API version is required on each SOAP request. This version number is the value of ns:version in https://www.paypal.com/wsdl/PalPalSvc.wsdl
24	For details about required and optional elements and values for specific requests, see the description of individual APIs.

Response Structure

The following is an annotated description of the structure of a SOAP response from the PayPal API where response is Success:

```xml
<?xml version="1.0"?>
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:cc="urn:ebay:apis:CoreComponentTypes"
    xmlns:wsu="http://schemas.xmlsoap.org/ws/2002/07/utility"
    xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
    xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/12/secext"
    xmlns:ebl="urn:ebay:apis:eBLBaseComponents"
    xmlns:ns="urn:ebay:api:PayPalAPI">
    <SOAP-ENV:Header>
        <Security
            xmlns="http://schemas.xmlsoap.org/ws/2002/12/secext"
            xsi:type="wsse:SecurityType"
        />
        <RequesterCredentials xmlns="urn:ebay:api:PayPalAPI"
            xsi:type="ebl:CustomSecurityHeaderType">
            <Credentials
                xmlns="urn:ebay:apis:eBLBaseComponents"
                xsi:type="ebl:UserIdPasswordType"
            />
        </RequesterCredentials>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body id="_0">
        <specific_api_name_Response xmlns="urn:ebay:api:PayPalAPI">
            <Timestamp xmlns="urn:ebay:api:PayPalAPI">
                dateTime_in_UTC/GMT
            </TIMESTAMP>
            <Ack xmlns="urn:ebay:apis:eBLBaseComponents">Success</Ack>
            <Version xmlns="urn:ebay:apis:eBLBaseComponents">
                serviceVersion
            </Version>
            <CorrelationId xmlns="urn:ebay:apis:eBLBaseComponents">
                applicationCorrelation
            </CorrelationID>
            <Build xmlns="urn:ebay:apis:eBLBaseComponents">
                api_build_number
            </Build>
            <elements_for_specific_api_response>
                data
            </elements_for_specific_api_response>
        </specific_api_name_Response>
    </SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```

*TABLE **1.6** Annotation of Generic SOAP Response*

Lines	Comment
22 and 31	The specific_api_name_Response start and end elements.
23	Each API response contains a timestamp with its date and time in UTC/GMT.
24	The `<Ack>` element contains the string `Success` after the corresponding request has been successfully processed. (For a complete list of possible returned strings, see "AbstractResponseType" on page 28.) In the case of errors, Ack is set to a value other than `Success`, and the response body contains an `<Errors>` element with information to help you troubleshoot the cause of the error. See "Error Responses" on page 21.
26	The `<CorrelationID>` element contains information about the PayPal application that processed the request. Use the value of this element if you need to troubleshoot a problem with one of your requests.
27 through 30	The different PayPal APIs return different structures depending on their response definitions. For detailed information, see the description of the individual APIs. **N O T E :** Because a field is defined in the formal structure of an API response does not mean that that field is necessarily returned. Data are returned in a response only if PayPal has recorded data that corresponds to the field.

Error Responses

If a request is malformed or some other error, the body of the SOAP response contains an `<Errors>` element with other elements that can help you troubleshoot the cause of the error. The most important of these additional elements are as follows:

- `ShortMessage`

- `LongMessage`

- `ErrorCode`

For a description of the contents of these elements, see Appendix A, "Error Codes and Messages for All PayPal APIs."

The following example shows the error response if your API username and password do not match a legitimate API username and password on file with PayPal.

Example of SOAP Error Response: Bad Username or Password

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope details not shown>
    <SOAP-ENV:Header>... details not shown.</SOAP-ENV:Header>
    <SOAP-ENV:Body id="_0">
        <GetTransactionDetailsResponse xmlns="urn:ebay:api:PayPalAPI">
```

```
            <Timestamp xmlns="urn:ebay:apis:eBLBaseComponents">
                2005-02-09T21:51:26Z
            </Timestamp>
            <Ack xmlns="urn:ebay:apis:eBLBaseComponents">Failure</Ack>
            <Errors
                xmlns="urn:ebay:apis:eBLBaseComponents"
                xsi:type="ebl:ErrorType">
                <ShortMessage xsi:type="xs:string">
                    Authentication/Authorization Failed
                </ShortMessage>
                <LongMessage xsi:type="xs:string">
                    Username/Password is incorrect
                </LongMessage>
                <ErrorCode xsi:type="xs:token">10002</ErrorCode>
                <SeverityCode xmlns="urn:ebay:apis:eBLBaseComponents">
                    Error
                </SeverityCode>
            </Errors>
            <CorrelationID xmlns="urn:ebay:apis:eBLBaseComponents">
                debugging_info
            </CorrelationID>
            <Version xmlns="urn:ebay:apis:eBLBaseComponents">
                1.000000
            </Version>
            <Build xmlns="urn:ebay:apis:eBLBaseComponents">1.0006</Build>
        .. other elements in response.
        </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

CorrelationID for Reporting Problems to PayPal

The value returned in `CorrelationID` is important for PayPal to determine the precise cause of any error you might encounter. If you have to troubleshoot a problem with your requests, we suggest that you capture the value of `CorrelationID` so you can report it to PayPal.

PayPal SOAP API Definitions

The PayPal SOAP API comprises individual API definitions for specific business functions. As a foundation, the API relies on eBay Business Language (eBL) base and core components. The core eBL structures `AbstractRequestType` and `AbstractResponseType` are the basis of the SOAP request and response of each PayPal API. `AbstractResponseType` is also the framework for error messages common across all PayPal APIs.

PayPal has made some schema design decisions that can affect how businesses design their own applications.

- Enumerations: Enumerations are defined directly in the PayPal API schema.

- Troubleshooting information: The PayPal API returns information about elements that trigger errors.

- Backward compatibility: The PayPal API is versioned so that business applications are backward compatible when new elements are introduced to the server-side schema.

N O T E : eBL defines many structures that are specific to processing auctions. PayPal's SOAP schema includes these definitions to maintain compatibility with eBay's SOAP and for possible future joint use of SOAP across both eBay and PayPal. The material in this book focuses only on those SOAP definitions pertinent to use of the PayPal SOAP API.

Character Encoding, Data Types and Formats, and Currencies

This section details allowed character encoding and character sets, date data types, and formats.

UTF-8 Character Encoding

The PayPal SOAP API service assumes that all data in SOAP requests is in Unicode, specifically, the Unicode (or UCS) Transformation Format, 8-bit encoding form (UTF-8).

In SOAP responses, the service always returns data in UTF-8.

Date/Time Formats

The PayPal SOAP API schema defines date/time values as Coordinated Universal Time (UTC/GMT), using ISO 8601 format, and of type `ns:dateTime`. An example date/tme stamp is 2006-08-24T05:38:48Z

Core Currency Amount Data Type

The core currency amount data type is call BasicAmountType and is derived from string, and all currency amount fields have the following structure:

1. The `currencyID` attribute is required.

2. The amount must have two decimal places.

3. The decimal separator must be a period ("`.`").

4. You must not use any thousands separator.

5. `BasicAmountType` has a data type of `ebl:CurrencyCodeType`, which defines a large number of different currency codes. However, for your processing to succeed, you must set *currencyCode* to one of the values from the first column of "" on page 23. Some APIs support only a subset of these currencies.

Here is an example. (The field name `Amount` is an example; actual field names can vary depending on the specific API.)

```
<Amount currencyID="currencyCode">3.00</Amount>
```

API Overview By Business Function

The PayPal SOAP API consists of individual APIs for specific business uses.

TABLE 1.8 API Overview by Business Function

Business Function	SOAP Request/Response
Authorization & Capture API	DoCaptureRequest DoCaptureResponse DoAuthorizationRequest DoAuthorizationResponse DoVoidRequest DoVoidResponse DoReauthorizationReques DoReauthorizationResponse
Direct Payment API	DoDirectPaymentRequest DoDirectPaymentResponse
Express Checkout API	SetExpressCheckoutRequest SetExpressCheckoutResponse GetExpressCheckoutDetailsRequest GetExpressCheckoutDetailsResponse DoExpressCheckoutPaymentRequest DoExpressCheckoutPaymentResponse
GetTransaction Details API	GetTransactionDetailsRequest GetTransactionDetailsResponse
MassPay API	MassPayRequest MassPayResponse
Refund Transaction API	RefundTransactionRequest RefundTransactionResponse
Transaction Search API	TransactionSearchRequest TransactionSearchResponse

APIs to Exchange Funds or Display Information

All PayPal APIs deal with business transactions. However, the APIs can be further categorized as follows:

- APIs that result in an exchange of funds:
- Authorization & Capture
- Direct Payment

- Express Checkout

- MassPay

- RefundTransaction

- APIs that provide information about business transactions:

- GetTransactionDetails

- TransactionSearch

Instant Payment Notification and the APIs

Use of a PayPal API that results in an exchange of funds triggers the sending of an Instant Payment Notification. The issuer of the API call is the one who receives the IPN. The APIs that send Instant Payment Notification are as follows. With the exception of MassPay, the API response contains all information you need about a transaction

- DoExpressCheckoutPayment

- DoDirectPayment

- DoCapture

- RefundTransaction

- MassPay

For more information about Instant Payment Notification, see the *PayPal Order Management Integration Guide*.

April 2007 *SOAP API Developer Reference*

2 Core eBL Request/Response Structures

The PayPal SOAP API is built on core structures from the eBL schema that define required elements in all SOAP requests and responses:

- `AbstractRequestType`
- `AbstractResponseType`

Diagram of core eBL Types

The following diagram represents the composition of the `AbstractRequestType` and `AbstractResponseType` elements.

FIGURE 2.1 Core eBL Types
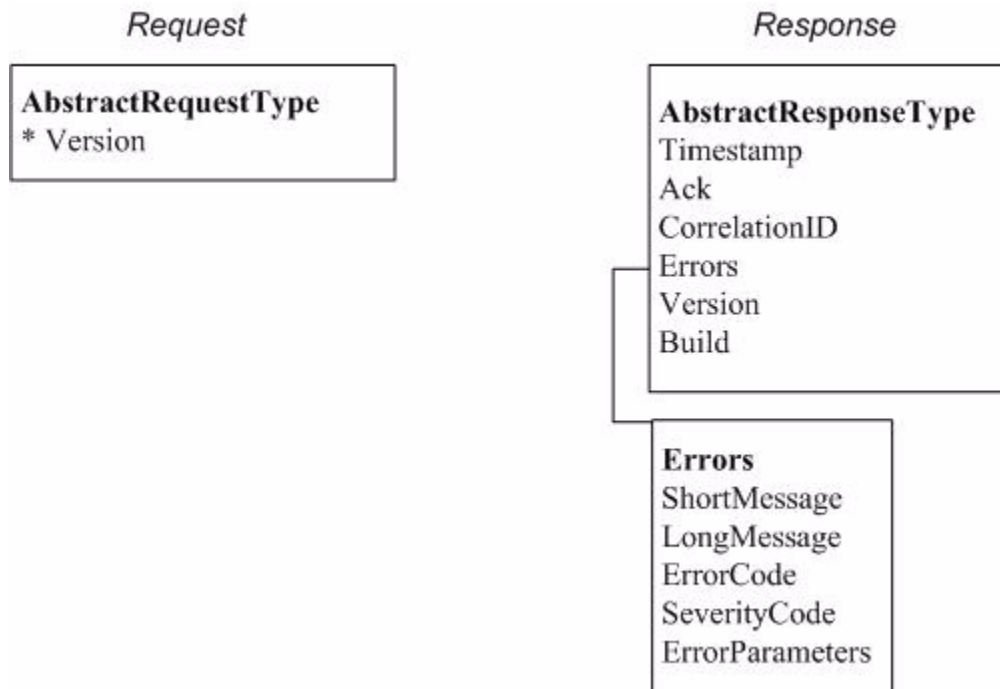
AbstractRequestType

`AbstractRequestType` is the base type definition of all PayPal SOAP requests.

I M P O R T A N T : You must set the value of the Version element for all requests to any PayPal API.

Fields

TABLE 2.1 AbstractRequestType Fields

Element	Description	Data Type	Allowable Values	Required?
Version	A string representing the version of the PayPal Schema., specifically, the value the version attribute of the wsdl:definitions element . For the location of the PayPal Schema, see "PayPal WSDL/XSD Schema Definitions" on page 14.	xs:string	See the the PayPal Schema.	Yes

AbstractResponseType

AbstractResponseType is the base type definition of all PayPal SOAP responses.

Fields

For error codes and messages and their meanings, see Appendix A, "Error Codes and Messages for All PayPal APIs."

TABLE 2.2 AbstractResponseType Fields

Element	Description	Data Type	Possible Values
Timestamp	The date and time (UTC/GMT) the response was generated by the PayPal SOAP service as a result of processing a request.	xs:dateTime	Transaction-specific

TABLE 2.2 AbstractResponseType Fields

Element	Description	Data Type	Possible Values

Ack	A token representing the application-level acknowledgement code. Contains one of the following enumerated values:
	• `Success`: Request processing succeeded
	• `Failure`: Request processing failed
	• `SuccessWithWarning`: Request processing completed successfully but with some warning information in `Errors` that could be useful for the requesting application to process or record.
	• `FailureWithWarning`: Request processing failed with some error and warning information that the requesting application should process to determine causes of failure.
	• `CustomCode`: Reserved for internal or future use.
		xs:token	`Success`
			`Failure`
			`SuccessWith`
			`Warning`
			`FailureWith`
			`Warning`
			`CustomCode`

Correlation ID	A string that identifies the exact API request that produced this response Character length: 41 characters maximum.	xs:string	Transaction-specific
Errors	Error code and messages for debugging a response. • `ErrorCode`: token defining the error code number • `ShortMessage`: string containing the error message • `LongMessage`: string describing the error message • `ErrorParameters`: error parameters from the credit card processor. Returned only with `DoDirectPaymentResponse`. • `SeverityCode`: returns `Error` or `Warning`.	ns:ErrorType xs:token xs:string xs:string ns:SeverityCodeType	Transaction-specific
Version	A string representing the version of the PayPal Schema, specifically, the value of the `version` attribute of the `wsdl:definitions` element . For the location of the PayPal Schema, see "PayPal WSDL/XSD Schema Definitions" on page 14.	xs:string	See the the PayPal Schema.
Build	A string representing the specific software build that processes the request and generates the response	xs:string	

3 Authorization & Capture API

The Authorization & Capture API consists of individual requests and responses, as shown below.

TABLE 3.1 Authorization & Capture API Names, Purposes, and Types of Authorization

API	Purpose	Used With Type of Authorization
DoCapture	Settle an order or previously authorized transaction and obtain payment for either the complete amount or any portion of it.	Order and Basic
DoAuthorization	Authorize an order that can be fulfilled over 29 days.	Order only
DoVoid	Void an original authorization or order	Order and Basic
DoReauthorization	Reauthorize a previously authorized transaction	Basic only

NOTE: For more information, see the chapter on Authorization and Capture in the *Express Checkout Integration Guide* or *Website Payment Pro Integration Guide*.

DoCapture

`DoCaptureRequest` is your request to completely or partially settle an order, an authorization, or reauthorization.

NOTE: `DoCaptureRequest` is for use with both basic and order authorizations.

Diagram of DoCapture Types

The following diagram represents the composition of the `DoCaptureRequestType` and `DoCaptureResponseType` elements. Elements required in the request are marked with an asterisk.

FIGURE 3.1 DoCapture Types
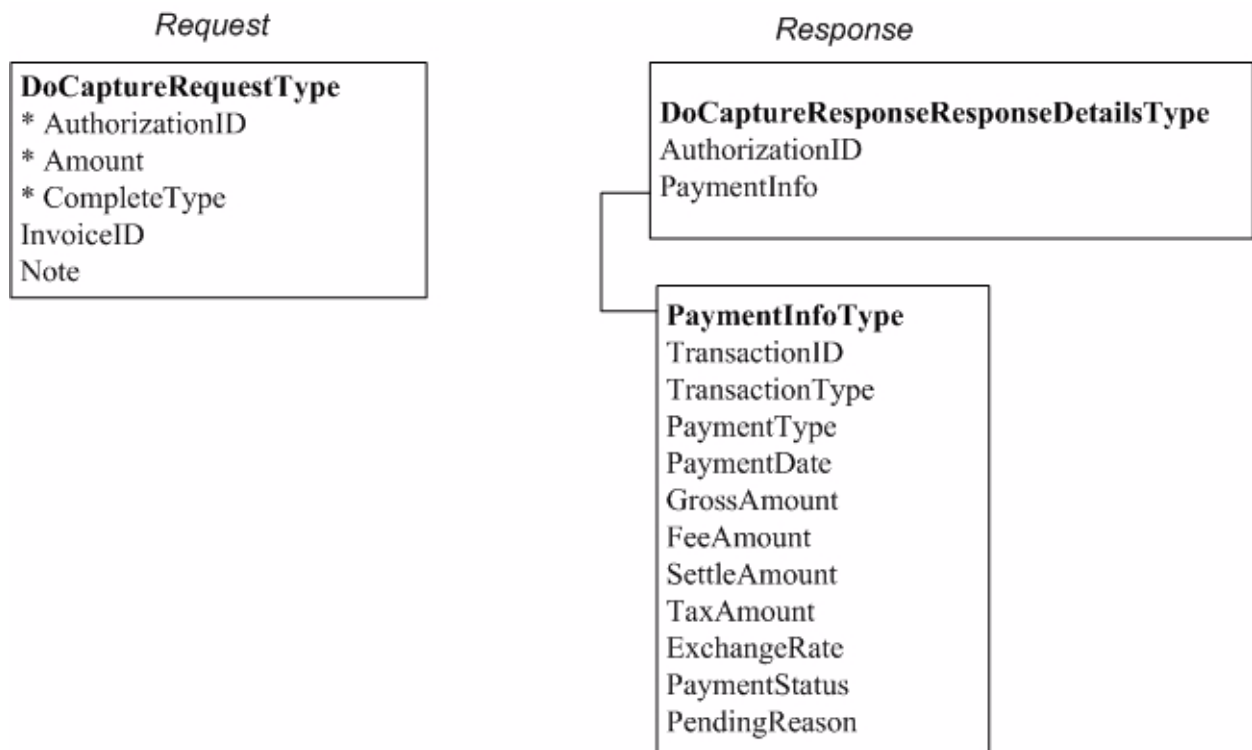
DoCaptureRequest

Request to capture funds from a PayPal member's account.

Fields

TABLE 3.2 DoCaptureRequest Fields

Field	Description	Data Type	Allowable Values	Required ?
Authorizati onID	The authorization identification number of the payment you want to capture. This is the transaction id returned from `DoExpressCheckoutPayment` or `DoDirectPayment`. Character length and limits: 19 single-byte characters maximum.	xs:string	See description.	Yes

TABLE 3.2 DoCaptureRequest Fields

Field	Description	Data Type	Allowable Values	Required?
Amount	Amount to capture. Limitations: Value is a positive number which cannot exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,).	ebl:BasicAmount Type	See description.	Yes
Complete Type	The value Complete indicates that this the last capture you intend to make. The value NotComplete indicates that you intend to make additional captures. **NOTE:** If Complete, any remaining amount of the original authorized transaction is automatically voided and all remaining open authorizations are voided. Character length and limits: 12 single-byte alphanumeric characters	ebl:Complete CodeType	Complete NotComplete	Yes
InvoiceID	Your invoice number or other identification number that is displayed to the merchant and customer in his transaction history. **NOTE:** This value on DoCapture will overwrite a value previously set on DoAuthorization. **NOTE:** The value is recorded only if the authorization you are capturing is an order authorization, not a basic authorization. Character length and limits: 127 single-byte alphanumeric characters	xs:string	See description.	No
Note	An informational note about this settlement that is displayed to the payer in email and in his transaction history. Character length and limits: 255 single-byte characters	xs:string	See description.	No

DoCaptureResponse

Response to DoCaptureRequest.

Fields

TABLE 3.3 DoCaptureResponse Fields

Field	Description	Data Type	Possible Values
Authorizati onID	The authorization identification number you specified in the request. Character length and limits: 19 single-byte characters maximum	xs:string	See description.
PaymentInfo	Information about the payment.	ebl:PaymentInf oType	See Table 3.4 on page 34.

TABLE 3.4 Response: PaymentInfoType Fields

Field	Description	Data Type	Possible Values
Transaction ID	Unique transaction ID of the payment. Character length and limitations: 17 single-byte characters	xs:string	Transaction-specific
ParentTrans actionID	Parent or related transaction identification number. This field is populated for the following transaction types: ● Reversal. Capture of an authorized transaction. ● Reversal. Reauthorization of a transaction. ● Capture of an order. The value of ParentTransactionID is the original OrderID. ● Authorization of an order. The value of ParentTransactionID is the original OrderID. ● Capture of an order authorization. ● Void of an order. The value of ParentTransactionID is the original OrderID. Character length and limits: 16 digits in xxxx-xxxx-xxxx-xxxx format		See description
ReceiptID	Receipt identification number Character length and limits: 16 digits in xxxx-xxxx-xxxx-xxxx format	xs:string	See description
Transaction Type	The type of transaction ● cart ● express-checkout Character length and limitations: 15 single-byte characters	ns:Payment TransactionCod e Type	See description
PaymentType	Indicates whether the payment is instant or delayed. Character length and limitations: Seven single-byte characters	ebl:PaymentCod eType	none echeck instant

TABLE 3.4 Response: PaymentInfoType Fields

Field	Description	Data Type	Possible Values
PaymentDate	Time/date stamp of payment. For example: 2006-08-15T17:23:15Z.	xs:dateTime	Transaction-specific
GrossAmount	The final amount charged, including any shipping and taxes from your Merchant Profile.	ebl:BasicAmountType	Transaction-specific
FeeAmount	PayPal fee amount charged for the transaction	ebl:BasicAmountType	Transaction-specific
SettleAmount	Amount deposited in your PayPal account if there is a currency conversion.	ebl:BasicAmountType	Transaction-specific
TaxAmount	Tax charged on the transaction, if any	ebl:BasicAmountType	Transaction-specific
ExchangeRate	Exchange rate if a currency conversion occurred. Relevant only if you are billing in the customer's non-primary currency. If the customer chooses to pay with a currency other than the non-primary currency, the conversion occurs in the customer's account. Character length and limitations: a decimal multiplier	xs:string	Transaction-specific

TABLE 3.4 Response: PaymentInfoType Fields

Field	Description	Data Type	Possible Values
PaymentStatus	Status of the payment. The status of the payment: • None: No status • Canceled-Reversal: This means a reversal has been canceled. For example, you won a dispute with the customer, and the funds for the transaction that was reversed have been returned to you. • Completed: The payment has been completed, and the funds have been added successfully to your account balance. • Denied: You denied the payment. This happens only if the payment was previously pending because of possible reasons described for the PendingReason element. • Expired: the authorization period for this payment has been reached. • Failed: The payment has failed. This happens only if the payment was made from your customer's bank account. • Pending: The payment is pending. See the PendingReason field for more information. • Refunded: You refunded the payment. • Reversed: A payment was reversed due to a chargeback or other type of reversal. The funds have been removed from your account balance and returned to the buyer. The reason for the reversal is specified in the ReasonCode element. • Processed: A payment has been accepted. • Voided: An authorization for this transaction has been voided.	ebl:PaymentStatusCodeType	None Canceled-Reversal Completed Denied Expired Failed Pending Refunded Reversed Processed Voided

TABLE 3.4 Response: PaymentInfoType Fields

Field	Description	Data Type	Possible Values
PendingReason	**NOTE:** PendingReason is returned in the response only if PaymentStatus is Pending. The reason the payment is pending: • none: No pending reason • address: The payment is pending because your customer did not include a confirmed shipping address and your Payment Receiving Preferences is set such that you want to manually accept or deny each of these payments. To change your preference, go to the Preferences section of your Profile. • echeck: The payment is pending because it was made by an eCheck that has not yet cleared. • intl: The payment is pending because you hold a non-U.S. account and do not have a withdrawal mechanism. You must manually accept or deny this payment from your Account Overview. • multi-currency: You do not have a balance in the currency sent, and you do not have your Payment Receiving Preferences set to automatically convert and accept this payment. You must manually accept or deny this payment. • verify: The payment is pending because you are not yet verified. You must verify your account before you can accept this payment. • other: The payment is pending for a reason other than those listed above. For more information, contact PayPal Customer Service.	ebl:PendingStatusCodeType	none address echeck intl multi-currency verify other

DoAuthorization

> **NOTE:** DoAuthorizationRequest is for use only with order authorizations, not basic authorizations.

DoAuthorizationRequest is your request to authorize a customer order that can be fulfilled within 29 days. You use DoAuthorizationRequest when you are ready to ship goods to your customer. After you ship, you can capture funds with DoCapture.

To use DoAuthorization you must have previously created a customer order. You can create an order in several ways:

● With the `DoExpressCheckoutPaymentRequest` or `DoDirectPaymentRequest` `PaymentAction` element set to `Order`.

● With a shopping cart or PayPal Website Payments transaction with the `paymentaction` HTML variable set to `Order`.

Diagram of DoAuthorization Types

The following diagram represents the composition of the `DoAuthorizationRequestType` and `DoAuthorizationResponseType` elements. Elements required in the request are marked with an asterisk.

FIGURE 3.2 DoAuthorization Types
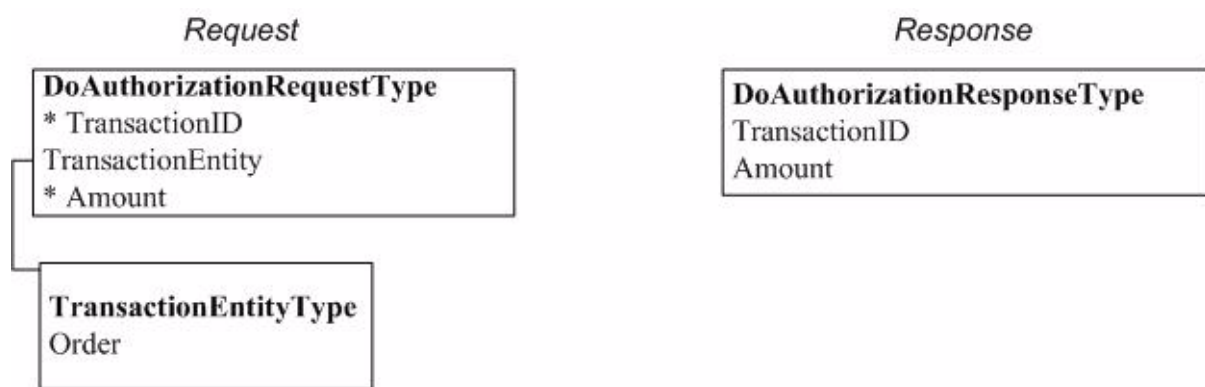
DoAuthorizationRequest

Request to authorize all or part of a customer order amount.

Fields

TABLE 3.5 DoAuthorizationRequest Fields

Field	Description	Data Type	Allowable Values	Required?
Transaction ID	The value of the order's transaction identification number returned by PayPal. Character length and limits: 19 single-byte characters maximum	xs:string	See description.	Yes

TABLE 3.5 DoAuthorizationRequest Fields

Field	Description	Data Type	Allowable Values	Required?
Amount	Amount to authorize. Limitations: Value is a positive number which cannot exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,).	`ebl:BasicAmountType`	See description.	Yes
Transaction Entity	Type of transaction to authorize. The only allowable value is `Order`, which means that the transaction represents a customer order that can be fulfilled over 29 days.	`ebl:TransactionEntityType`	`Order`	No

DoAuthorizationResponse

Response to `DoAuthorizationRequest`.

Fields

TABLE 3.6 DoAuthorizationResponse Fields

Field	Description	Data Type	Possible Values
Transaction ID	An authorization identification number. Character length and limits: 19 single-byte characters	`xs:string`	See description.
Amount	The amount you specified in the request.	`ebl:BasicAmountType`	See description.

DoVoid

`DoVoidRequest` voids an order or an authorization.

IMPORTANT: The `AuthorizationID` value on `DoVoidRequest` must be the original authorization identification number, not the value of `AuthorizationID` returned by `DoReauthorizationResponse`.

By definition, when the authorization period expires, the authorization or reauthorization of a transaction is implicitly voided.

NOTE: `DoVoidRequest` is for use with both basic and order authorizations.

Diagram of DoVoid Types

The following diagram represents the composition of the `DoVoidRequestType` and `DoVoidResponseType` elements. Elements required in the request are marked with an asterisk.

FIGURE 3.3 DoVoid Types

DoVoidRequest

Request to void a prior authorization.

Fields

TABLE 3.7 DoVoidRequest Fields

Field	Description	Data Type	Allowable Values	Required?
Authorizati onID	The value of the original authorization identification number returned by a PayPal product. **IMPORTANT:** If you are voiding a transaction that has been reauthorized, use the ID from the original authorization, and not the reauthorization. Character length and limits: 19 single-byte characters	xs:string	See description.	Yes
Note	An informational note about this void that is displayed to the payer in email and in his transaction history. Character length and limits: 255 single-byte characters	xs:string	See description.	No

DoVoidResponse

Response to `DoVoidRequest`.

Fields

TABLE 3.8 DoVoidResponse Fields

Field	Description	Data Type	Possible Values
Authorizati on ID	The authorization identification number you specified in the request. Character length and limits: 19 single-byte characters	xs:string	See description.

DoReauthorization

NOTE: DoReauthorization is for use only with basic authorizations, not order authorizations.

To use DoReauthorization you must have previously authorized a transaction and that transaction must have passed its settlement period. Calling DoReauthorization on a valid transaction will fail. You can authorize a transaction in several ways:

● From the Merchant Services tab of your PayPal merchant account.

● With the DoExpressCheckoutPaymentRequest PaymentAction element set to Authorization.

● With a shopping cart transaction (such as one that uses PayPal Website Payments) with the paymentaction HTML variable set to Authorization.

You can invoke DoReauthorization as many times as necessary to obtain one successful reauthorization. When DoReauthorization returns success, the settlement period restarts, you can capture funds, but you can no longer reauthorize.

Diagram of DoReauthorization Types

The following diagram represents the composition of the DoReauthorizationRequestType and DoReauthorizationResponseType elements. Elements required in the request are marked with an asterisk.

FIGURE 3.4 DoReauthorization Types

DoReauthorizationRequest

Request to use Authorization & Capture to reauthorize a transaction.

Fields

TABLE 3.9 DoReauthorizationRequest Fields

Field	Description	Data Type	Allowable Values	Required ?
Authorizati on ID	The value of a previously authorized transaction identification number returned by PayPal. Character length and limits: 19 single-byte characters maximum	xs:string	Any previously authorized PayPal transaction identification number.	Yes
Amount	Amount to reauthorize. Limitations: Value is a positive number which cannot exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,).	eBL:BasicAmoun tType	See description.	Yes

DoReauthorizationResponse

Response to DoReauthorizationRequest.

Fields

TABLE 3.10 DoReauthorizationResponse Fields

Field	Description	Data Type	Possible Values
Authorizati on ID	A new authorization identification number. Character length and limits:19 single-byte characters	xs:string	See description.

4

Direct Payment API

With the Direct Payment API, you can obtain payment through PayPal directly from a buyer's credit card without the buyer logging into PayPal.

Diagram of DoDirectPayment Types

The following diagram represents the composition of the `DoDirectPaymentRequestType` and `DoDirectPaymentResponseType` elements. Elements required in the request are indicated with an asterisk.

FIGURE 4.1 **DoDirectPayment Types**
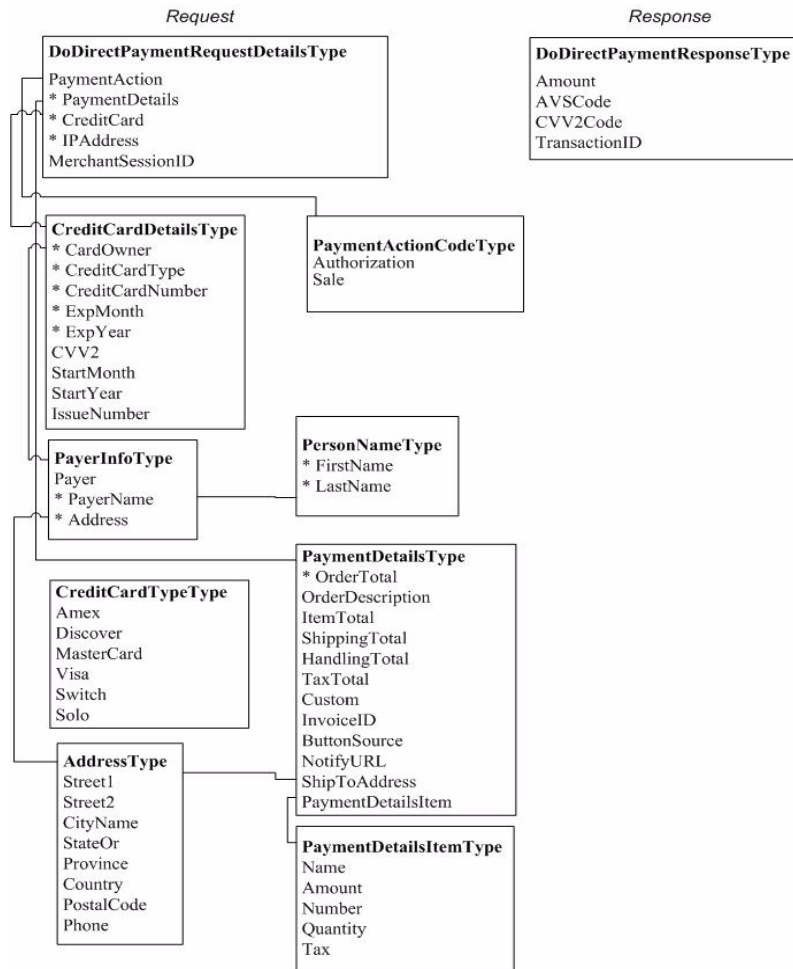
DoDirectPaymentRequest

Request to use PayPal Direct Payment to charge or authorize a credit card.

You must include the `PaymentAction` element in `DoDirectPaymentRequest`. Allowable values for the `PaymentAction` element are `Sale` or `Authorization`:

- Set `PaymentAction` to `Sale` if the buyer's order is final (for example, if the total cost of the order is complete with shipping, handling and tax charges). This transaction is your payment, and you expect no additional charge to the buyer.

- Set `PaymentAction` to `Authorization` if the total cost of the order at point of sale is not complete for some reason (such as unknown shipping charges) and you expect to capture funds after checkout is complete. You can then use PayPal Authorization & Capture to obtain payment.

In either case, besides the `PaymentAction` element, `DoDirectPaymentRequest` requires additional elements, such as the amount of the transaction, IP address of the buyer's browser, credit card number, type, and expiration date, and information about both the payer and the payment.

Fields

TABLE 4.1 DoDirectPaymentRequest Fields

Field	Description	Data Type	Allowable Values	Required ?
PaymentAction	How you want to obtain payment: • `Authorization` indicates that this payment is a basic authorization subject to settlement with PayPal Authorization & Capture. • `Sale` indicates that this is a final sale for which you are requesting payment. Character length and limit: Up to 13 single-byte alphabetic characters	ebl:PaymentActionCodeType	Authorization Default: Sale **NOTE:** Order is not allowed for Direct Payment.	No
CreditCard	Information about the credit card to be charged.	ebl:CreditCardDetailsType	See Table 4.2 on page 46.	Yes
PaymentDetails	Information about the payment	ebl:PaymentDetailsType	See Table 4.3 on page 47.	Yes
IPAddress	IP address of the payer's browser. **IMPORTANT:** PayPal records this IP addresses as a means to detect possible fraud. Character length and limitations: 15 single-byte characters, including periods, for example: `255.255.255.25`.	xs:string	Any valid Internet Protocol address.	Yes
MerchantSessionId	Your customer session identification token. **NOTE:** PayPal records this optional session identification token as an additional means to detect possible fraud. Character length and limitations: 64 single-byte numeric characters	xs:string		No

TABLE 4.2 CreditCardDetailsType Fields

Field	Description	Data Type	Allowable Values	Required?
CreditCardType	Type of credit card. Character length and limitations: Up to ten single-byte alphabetic characters. Allowable values: • Visa • MasterCard • Discover • Amex • Switch: See important note. • Solo: See important note. **IMPORTANT:** If the credit card type is Switch or Solo, the currencyId must be GBP. In addition, either StartMonth and StartYear or IssueNumber must be specified.	ebl:CreditCardType	See description.	Yes
CreditCardNumber	Credit card number Character length and limitations: numeric characters only. No spaces or punctutation. Must conform with modulo and length required by each credit card type.	xs:string	See description.	Yes
ExpMonth	Credit card expiration month Character length and limitations: Two single-byte numeric characters, including leading zero.	xs:int	See description.	Yes
ExpYear	Credit card expiration year Character length and limitations: Four single-byte numeric characters.	xs:int	See description.	Yes
CVV2	Card Verification Value, version 2. **NOTE:** Your Merchant Account settings determine whether this field is required. Contact your PayPal Account Manager for more information. Character length for Visa, MasterCard, and Discover: exactly three digits. Character length for American Express: exactly four digits. **IMPORTANT:** To comply with credit card processing regulations, once a transaction has been completed, you must not store the value of CVV2.	xs:string	See description.	See note in description.

TABLE 4.2 CreditCardDetailsType Fields

Field	Description	Data Type	Allowable Values	Required ?
CardOwner	Details about the owner of the credit card.	ns:PayerInfoType	See Table 4.6 on page 51.	Yes
StartMonth	Month that Switch or Solo card was issued. Character length: two-digit, zero-filled if necessary.	xs:int	See description.	No.
StartYear	Year that Switch or Solo card was issued. Character length: four digits.	xs:int	See description.	No.
IssueNumber	Issue number of Switch or Solo card. Character length: two numeric digits maximum.	xs:int	See description.	No.

TABLE 4.3 Request: PaymentDetailsType Fields

Field	Description	Data Type	Allowable Values	Required ?
OrderTotal	Total of order, including shipping, handling, and tax. Limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). The only valid currencies are as follows: • AUD: Australian Dollar • CAD: Canadian Dollar • EUR: Euro • GBP: Pound Sterling • JPY: Japanese Yen • USD: U.S. Dollar	ebl: BasicAmount Type	See description.	Yes
ItemTotal	Sum of cost of all items in this order. Limitations: The value must be a positive number and cannot exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). NOTE: You must specify this value if you specify Amount in PaymentDetailsItem or if you specify values for ShippingTotal or HandlingTotal.	ebl: BasicAmount Type	See description.	No

TABLE 4.3 Request: PaymentDetailsType Fields

Field	Description	Data Type	Allowable Values	Required ?
Shipping Total	Total shipping costs for this order. Limitations: The value must be zero or greater and cannot exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). The only valid currencies are as follows: • AUD: Australian Dollar • CAD: Canadian Dollar • EUR: Euro • GBP: Pound Sterling • JPY: Japanese Yen • USD: U.S. Dollar **NOTE:** If you specify a value for `ShippingTotal`, you must also specify a value for `ItemTotal`.	`ebl: BasicAmount Type`	See description.	No
Handling Total	Total handling costs for this order. Limitations: The value must be zero or greater and cannot exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). The only valid currencies are as follows: • AUD: Australian Dollar • CAD: Canadian Dollar • EUR: Euro • GBP: Pound Sterling • JPY: Japanese Yen • USD: U.S. Dollar **NOTE:** If you specify a value for `HandlingTotal`, you must also specify a value for `ItemTotal`.	`ebl: BasicAmount Type`	See description.	No

TABLE 4.3 Request: PaymentDetailsType Fields

Field	Description	Data Type	Allowable Values	Required?
TaxTotal	Sum of tax for all items in this order. Limitations: The value must be zero or greater and cannot exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). The only valid currencies are as follows: • AUD: Australian Dollar • CAD: Canadian Dollar • EUR: Euro • GBP: Pound Sterling • JPY: Japanese Yen • USD: U.S. Dollar	ebl: BasicAmount Type	See description.	No
Order Description	Description of items the customer is purchasing. Character length and limitations: 127 single-byte alphanumeric characters	xs:string	See description.	No
Custom	A free-form field for your own use. Character length and limitations: 256 single-byte alphanumeric characters	xs:string	See description.	No
InvoiceID	Your own invoice or tracking number. Character length and limitations: 127 single-byte alphanumeric characters	xs:string	See description.	No
Button Source	An identification code for use by third-party applications to identify transactions. Character length and limitations: 32 single-byte alphanumeric characters	xs:string	See description.	No
NotifyURL	Your URL for receiving Instant Payment Notification (IPN) about this transaction. **NOTE:** If you do not specify this URL in the request, the notification URL from your Merchant Profile is used, if one exists. Character length and limitations: 2,048 single-byte alphanumeric characters	xs:string	See description.	No
ShipTo Address	Address the order will be shipped to.	ebl: AddressType	See Table 4.4 on page 50.	No
Payment DetailsItem	Details about each individual item included in the order	ebl: PaymentDetails ItemType	See Table 4.5 on page 50.	No

> **IMPORTANT:** In the following table, any fields in the WSDL or XSD files that are not described here are ignored. For example, do not set `CountryName`.

TABLE *4.4 Request: ShipToAddress: AddressType Fields*

Field	Descriptions	Data Type	Allowable Values	Required?
Name	Person's name associated with this address. Character length and limitations: 32 single-byte characters	xs:string	See description.	Yes
Street1	First street address. Character length and limitations: 100 single-byte characters	xs:string	See description.	Yes
Street2	Second street address. Character length and limitations: 100 single-byte characters	xs:string	See description.	No
CityName	Name of city. Character length and limitations: 40 single-byte characters	xs:string	See description.	Yes
StateOr Province	State or province. Character length and limitations: 40 single-byte characters Required for US addresses only.	xs:string	See Appendix B, "State and Province Abbreviations."	No
PostalCode	U.S. ZIP code or other country-specific postal code. Character length and limitations: 20 single-byte characters	xs:string	See description.	Yes
Country	Country code. Character limit: Two single-byte characters	ebl:CountryCodeType	See Appendix F, "Country Codes.	Yes
Phone	Phone number. Character length and limit: 20 single-byte characters	xs:string	See description	No

TABLE *4.5 Request: PaymentDetailsItemType Fields*

Field	Description	Data Type	Allowable Values	Required?
Name	Item name. Character length and limitations: 127 single-byte characters	xs:string	See description.	No

TABLE 4.5 Request: PaymentDetailsItemType Fields

Field	Description	Data Type	Allowable Values	Required ?
Amount	Cost of item Limitations: Value can be positive, negative or zero and must not exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). **NOTE:** You must set the currencyID attribute to one of the three-character currency codes for any of the supported PayPal currencies. See "" on page 23. **NOTE:** If you specify this value, you must specify a value for `ItemTotal`.	`ebl:BasicAmountType`	See description.	No
Number	Item number. Character length and limitations: 127 single-byte characters	`xs:string`	See description.	No
Quantity	Item quantity. Character length and limitations: Any positive integer	`xs:string`	See description.	No
Tax	Item sales tax. Limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). Currency code is set the same as for `OrderTotal`.	`ebl:BasicAmountType`	See description.	No

TABLE 4.6 CardOwner: PayerInfoType Fields

Field	Description	DataType	Allowable Values	Required ?
Payer	Email address of payer. Character length and limitations: 127 single-byte characters	`ns:EmailAddressType`	See description.	No
FirstName	Payer's first name. Character length and limitations: 25 single-byte characters	`ns:PersonNameType`	See description.	Yes
LastName	Payer's last name. Character length and limitations: 25 single-byte characters	`ns:PersonNameType`	See description.	Yes

TABLE *4.6 CardOwner: PayerInfoType Fields*

Field	Description	DataType	Allowable Values	Required?
Address	Payer's billing address information	ns:AddressType	See Table 4.7 on page 52.	Yes

TABLE *4.7 CardOwner: AddressType Fields*

Field	Description	DataType	Allowable Values	Required?
Street1	First street address. Character length and limitations: 100 single-byte characters	xs:string	See description.	Yes
Street2	Second street address. Character length and limitations: 100 single-byte characters	xs:string	See description.	No
CityName	Name of city. Character length and limitations: 40 single-byte characters	xs:string	See description.	Yes
StateOr Province	State or province. Character length and limitations: 40 single-byte characters Required for US addresses only.	xs:string	See Appendix B, "State and Province Abbreviations."	No
Country	Country code. Character limit: Two single-byte characters	ebl:CountryCode Type xs:string	See Appendix F, "Country Codes.	Yes
PostalCode	U.S. ZIP code or other country-specific postal code. Character length and limitations: 20 single-byte characters	xs:string	See description.	Yes
Phone	Phone number. Character length and limit: 20 single-byte characters	xs:string	See description.	No

DoDirectPaymentResponse

Response to DoDirectPaymentRequest.

The Direct Payment API response contains the TransactionID element. In the case of a final Sale, TransactionID is your record of the transaction processed with PayPal. In the

case of an `Authorization`, `TransactionID` is your key to capturing funds with PayPal Authorization & Capture.

`DoDirectPaymentResponse` also returns the result of checking the buyer's billing address and credit card number (as submitted by you with `DoDirectPaymentRequest`) against the U.S. banking industry standard.

Address Verification System (AVS) and Credit Card Verification (CVV2) system.

Fields

TABLE 4.8 DoDirectPaymentResponse Fields

Field	Description	Data Type	Possible Values
Transaction ID	Unique transaction ID of the payment. **NOTE:** If the `PaymentAction` of the request was `Authorization`, the value of TransactionID is your AuthorizationID for use with the Authorization & Capture APIs. Character length and limitations: 19 single-byte characters	xs:string	See description.
Amount	The amount of the payment as specified by you on `DoDirectPaymentRequest`.	ebl:BasicAmount Type	See description.
AVSCode	Address Verification System response code. Character limit: One single-byte alphanumeric character	xs:string	See "AVS Response Codes" on page 53.
CVV2Code	Result of the CVV2 check by PayPal.	xs:string	See "CVV2 Response Codes" on page 55.

AVS Response Codes

The following tables list the AVS response codes.

TABLE 4.9 AVS Response Codes for Visa, MasterCard, Discover, and American Express

AVS Code	Meaning	Matched Details
A	Address	Address only (no ZIP)

TABLE 4.9 AVS Response Codes for Visa, MasterCard, Discover, and American Express

AVS Code	Meaning	Matched Details
B	International "A"	Address only (no ZIP)
C	International "N"	None **NOTE:** The transaction is declined.
D	International "X"	Address and Postal Code
E	Not allowed for MOTO (Internet/Phone) transactions	Not applicable **NOTE:** The transaction is declined.
F	UK-specific "X"	Address and Postal Code
G	Global Unavailable	Not applicable
I	International Unavailable	Not applicable
N	No	None **NOTE:** The transaction is declined.
P	Postal (International "Z")	Postal Code only (no Address)
R	Retry	Not applicable
S	Service not Supported	Not applicable
U	Unavailable	Not applicable
W	Whole ZIP	Nine-digit ZIP code (no Address)
X	Exact match	Address and nine-digit ZIP code
Y	Yes	Address and five-digit ZIP
Z	ZIP	Five-digit ZIP code (no Address)
All others	Error	Not applicable

TABLE 4.10 AVS Response Codes for Switch and Solo

0	All the address information matched.	All information matched
1	None of the address information matched.	None **NOTE:** The transaction is declined.
2	Part of the address information matched.	Partial

TABLE 4.1O AVS Response Codes for Switch and Solo

3	The merchant did not provide AVS information. Not processed.	Not applicable
4	Address not checked, or acquirer had no response. Service not available.	Not applicable
Null	No AVS response was obtained. Default value of field.	Not applicable

CVV2 Response Codes

The following tables list the CVV2 response codes.

TABLE 4.11 CVV2 Response Codes for Visa, MasterCard, Discover, and American Express

CVV2 Code	Meaning	Matched Details
M	Match	CVV2
N	No match	None
P	Not processed	Not applicable
S	Service not supported	Not applicable
U	Service not available	Not applicable
X	No response	Not applicable

TABLE 4.12 CVV2 Response Codes for Switch and Solo

0	Matched	CVV2
1	No match	None
2	The merchant has not implemented CVV2 code handling	Not applicable
3	Merchant has indicated that CVV2 is not present on card	Not applicable
4	Service not available	Not applicable
All others	Error	Not applicable

5

Express Checkout API

The Express Checkout API is the Website Payments Pro alternative to standard PayPal shopping cart processing.

IMPORTANT: For a full understanding of the complete details about how to use Express Checkout, consult the PayPal Express Checkout Integration Guide.

The Express Checkout API consists of three distinct SOAP requests that you make at the three integration points during a customer checkout. The customer is also redirected to PayPal. Here is the sequence of steps:

1. `SetExpressCheckout`

2. Redirect the customer's browser to the PayPal Login page

3. `GetExpressCheckoutDetails`

4. `DoExpressCheckoutPayment`

IMPORTANT: All currency amount fields in the three Express Checkout APIs require that the `currencyID` attribute be set to one of the supported currency codes. For any one particular buyer's checkout session, do not mix currencies either in any single SOAP request or across SOAP requests. For example, do not set `currencyID` to USD in one request and then to GBP in a later request for the same buyer.

Express Checkout for eBay Auctions

For using Express Checkout with eBay Auctions, you have some additional fields you can set to control the behavior of Express Checkout:

1. On SetExpressCheckout:

    – ChannelType of eBayItem

2. On DoExpressCheckoutPayment:

    – PaymentAction must be Sale
    – EbayItemPaymentDetailsItem.ItemNumber
    – EbayItemPaymentDetailsItem.AuctionTransactionID
    – EbayItemPaymentDetailsItem.OrderID

SetExpressCheckout

`SetExpressCheckout` indicates to PayPal that you are using Express Checkout to obtain payment from your customer.

Diagram of Set ExpressCheckout Types

The following diagram represents the composition of the `SetExpressCheckoutRequestType` and `SetExpressCheckoutResponseType` elements. Elements required in the request are indicated with an asterisk.

FIGURE 5.1 **SetExpressCheckout Types**
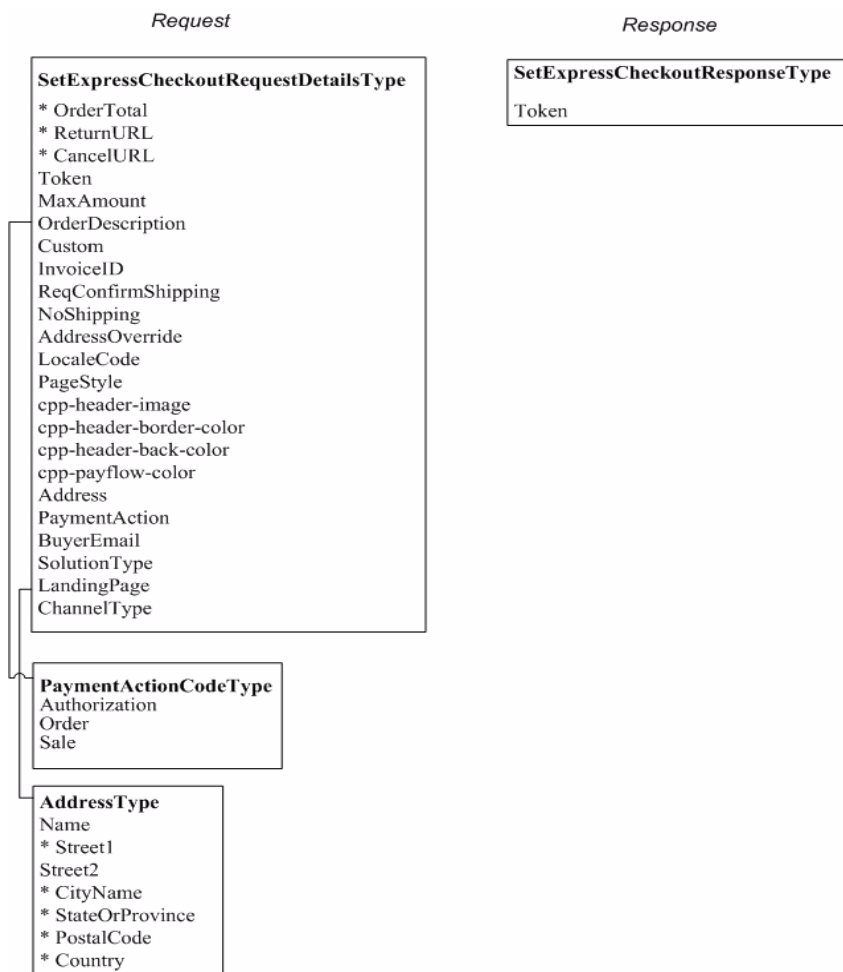
SetExpressCheckoutRequest

Request to use PayPal Express Checkout to obtain payment.

IMPORTANT: Any fields in the WSDL or XSD files that are not described here are ignored. For example, do not set CountryName.

Fields

TABLE 5.1 SetExpressCheckoutRequest Fields

Field	Description	Data Type	Allowable Values	Required?
Token	A timestamped token, the value of which was returned by SetExpressCheckout response. Character length and limitations: 20 single-byte characters	`ebl:Express CheckoutTokenType`	See the description of Table 5.3, "SetExpressCheckoutResponse Fields."	No
OrderTotal	The total cost of the order to the customer. If shipping cost and tax charges are known, include them in this value; if not, this value should be the current sub-total of the order. **NOTE:** Limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). **NOTE:** You must set the currencyID attribute to one of the three-character currency codes for any of the supported PayPal currencies.	`cc:BasicAmountType`	See description.	Yes
MaxAmount	The expected maximum total amount of the complete order, including shipping cost and tax charges. **NOTE:** Limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). **NOTE:** You must set the currencyID attribute to one of the three-character currency codes for any of the supported PayPal currencies.	`cc:BasicAmountType`	See description.	No

TABLE 5.1 SetExpressCheckoutRequest Fields

Field	Description	Data Type	Allowable Values	Required ?
Order Description	Description of items the customer is purchasing. Character length and limitations: 127 single-byte alphanumeric characters	xs:string	See description.	No
Custom	A free-form field for your own use, such as a tracking number or other value you want PayPal to return on `GetExpressCheckoutDetails` response and response. Character length and limitations: 256 single-byte alphanumeric characters	xs:string	See description.	No
InvoiceID	Your own unique invoice or tracking number. PayPal returns this value to you on response. Character length and limitations: 127 single-byte alphanumeric characters	xs:string	See description.	No
ReturnURL	URL to which the customer's browser is returned after choosing to pay with PayPal. **IMPORTANT:** PayPal recommends that the value be the final review page on which the customer confirms the order and payment or billing agreement. Character length and limitations: no limit.	xs:string	See description.	Yes

TABLE 5.1 SetExpressCheckoutRequest Fields

Field	Description	Data Type	Allowable Values	Required ?
CancelURL	URL to which the customer is returned if he does not approve the use of PayPal to pay you. **IMPORTANT:** PayPal recommends that the value be the original page on which the customer chose to pay with PayPal or establish a billing agreement. Character length and limitations: no limit	xs:string	See description.	Yes
Address	Customer's shipping address. **NOTE:** If you include a shipping address and set the AddressOverride element on the request, PayPal returns this same address in GetExpressCheckoutDetails Response.	ebl:AddressType	See Table 5.2 on page 64."	No
ReqConfirm Shipping	The value 1 indicates that you require that the customer's shipping address on file with PayPal be a confirmed address. **NOTE:** Setting this field overrides the setting you have specified in your Merchant Account Profile. Character length and limitations: One single-byte numeric character.	xs:string	0 1	No
NoShipping	The value 1 indicates that on the PayPal pages, no shipping address fields should be displayed whatsoever. Character length and limitations: Four single-byte numeric character.	xs:string	0 1	No

TABLE 5.1 SetExpressCheckoutRequest Fields

Field	Description	Data Type	Allowable Values	Required ?
Address Override	The value 1 indicates that the PayPal pages should display the shipping address set by you in this `SetExpressCheckout` request, not the shipping address on file with PayPal for this customer. **NOTE:** Displaying the PayPal street address on file does not allow the customer to edit that address.	xs:string	0 1	No
LocaleCode	Locale of pages displayed by PayPal during Express Checkout. Character length and limitations: Any two-character country code. The following two-character country codes are supported by PayPal: • AU • DE • FR • IT • GB • ES • US Any other value will default to US. **NOTE:** For the list of country codes, see Appendix F, "Country Codes."	xs:string	See description.	No
PageStyle	Sets the Custom Payment Page Style for payment pages associated with this button/link. This value corresponds to the HTML variable page_style for customizing payment pages. The value is the same as the Page Style Name you chose when adding or editing the page style from the `Profile` subtab of the `My Account` tab of your PayPal account. Character length and limitations: 30 single-byte alphabetic characters.	xs:string	See description.	No
cpp-header-image	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. PayPal recommends that you provide an image that is stored on a secure (https) server. Character length and limit: 127 single-byte alphanumeric characters	xs:string	See description.	No

TABLE 5.1 SetExpressCheckoutRequest Fields

Field	Description	Data Type	Allowable Values	Required ?
cpp-header-border-color	Sets the border color around the header of the payment page. The border is a 2-pixel perimeter around the header space, which is 750 pixels wide by 90 pixels high. Character length and limitations: Six character HTML hexadecimal color code in ASCII	xs:string	See description.	No
cpp-header-back-color	Sets the background color for the header of the payment page. Character length and limitation: Six character HTML hexadecimal color code in ASCII	xs:string	See description.	No
cpp-payflow-color	Sets the background color for the payment page. Character length and limitation: Six character HTML hexadecimal color code in ASCII	xs:string	See description.	No

PaymentAction	How you want to obtain payment:
	• `Authorization` indicates that this payment is a basic authorization subject to settlement with PayPal Authorization & Capture.
	• `Order` indicates that this payment is an order authorization subject to settlement with PayPal Authorization & Capture.
	• `Sale` indicates that this is a final sale for which you are requesting payment.
	
	NOTE: You cannot set this value to `Sale` on `SetExpressCheckout` request and then change this value to `Authorization` on the final API `DoExpressCheckoutPayment` request.
	
	Character length and limit: Up to 13 single-byte alphabetic characters
		ebl:PaymentAction CodeType
		
		xs:token
			Default: The transaction resulting from `DoExpress Checkout Payment Request` will be a final sale.
			
			`Authorization`
			`Order`
			`Sale`
				No

BuyerEmail	Email address of the buyer as entered during checkout. PayPal uses this value to pre-fill the PayPal membership sign-up portion of the PayPal login page. Character length and limit: 127 single-byte alphanumeric characters	ebl:EmailAddress Type	See description.	No
SolutionType	Type of checkout flow: • Sole: Express Checkout for auctions • Mark: normal Express Checkout	eBL:SolutionType peType	Sole Mark	No

TABLE 5.1 SetExpressCheckoutRequest Fields

Field	Description	Data Type	Allowable Values	Required?
LandingPage	Type of PayPal page to display: • Billing: non-PayPal account • Login: PayPal account login	`eBL:LandingPageType`	Billing Mark	No
ChannelType	Type of channel: • Merchant: non-auction seller • eBayItem: eBay auction	`eBL:ChannelType`	Merchant eBayItem	No

NOTE: The `AddressType` elements are required only if you include the `Address` element in the SOAP request.

IMPORTANT: Any fields in the WSDL or XSD files that are not described here are ignored. For example, do not set CountryName.

TABLE 5.2 Address (Shipping Address): AddressType Elements

Field	Descriptions	Data Type	Allowable Values	Required?
Name	Person's name associated with this shipping address. Character length and limitations: 32 single-byte characters	`xs:string`	See description.	Yes
Street1	First street address. Character length and limitations: 100 single-byte characters	`xs:string`	See description.	Yes
Street2	Second street address. Character length and limitations: 100 single-byte characters	`xs:string`	See description.	No
CityName	Name of city. Character length and limitations: 40 single-byte characters	`xs:string`	See description.	Yes
StateOr Province	State or province. Character length and limitations: 40 single-byte characters Required for US addresses only.	`xs:string`	See Appendix E, "State and Province Abbreviations."	No
PostalCode	U.S. Zip code or other country-specific postal code. Character length and limitations: 20 single-byte characters	`xs:string`	See description.	Yes

TABLE 5.2 Address (Shipping Address): AddressType Elements

Field	Descriptions	Data Type	Allowable Values	Required?
Country	Country code. Character limit: Two single-byte characters.	ebl:Country Code Type xs:string	See Appendix F, "Country Codes."	Yes
Phone	Phone number. Character length and limit: 20 single-byte characters	xs:string	See description.	No

SetExpressCheckoutResponse

Response to SetExpressCheckoutRequest.

Fields

TABLE 5.3 SetExpressCheckoutResponse Fields

Field	Description	Data Type	Value
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. If you set the token in the SetExpressCheckout request, the value of the token in the response is identical to the value in the request. Character length and limitations: 20 single-byte characters	xs:string	See description.

GetExpressCheckoutDetails

GetExpressCheckoutDetails returns information about the customer, including name and address on file with PayPal.

Diagram of GetExpressCheckoutDetails Types

The following diagram represents the composition of the GetExpressCheckoutDetailsRequestType and GetExpressCheckoutDetailsResponseType elements.

Elements required in the request are indicated with an asterisk.

FIGURE 5.2 GetExpressCheckout Types
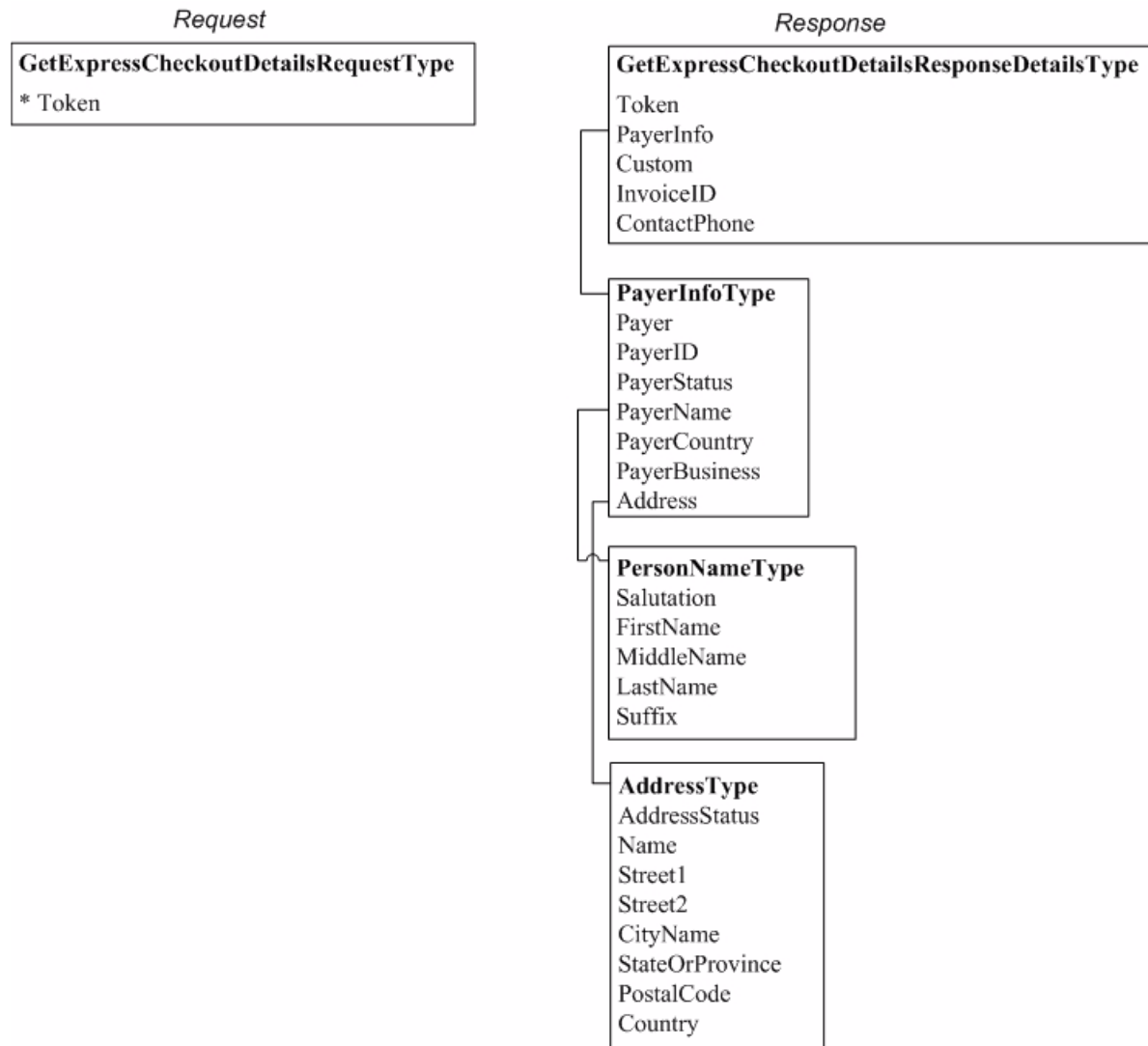
GetExpressCheckoutDetailsRequest

Obtain details about a payment to be processed by PayPal Express Checkout.

Fields

TABLE 5.4 GetExpressCheckoutDetailsRequest Fields

Field	Description	Data Type	Allowable Values	Required?
Token	A timestamped token, the value of which was returned by `SetExpressCheckout` response. Character length and limitations: 20 single-byte characters	`xs:string`	An unexpired token	Yes

GetExpressCheckoutDetailsResponse

Response to `GetExpressCheckoutDetailsRequest`.

Fields

TABLE 5.5 GetExpressCheckoutDetailsResponse Fields

Field	Description	Data Type	Possible Values
Token	The timestamped token value that was returned by `SetExpressCheckout` response and passed on `GetExpressCheckoutDetails` request. Character length and limitations: 20 single-byte characters	`xs:string`	See the description of Table 5.3, "SetExpressCheckoutResponse Fields."
PayerInfo	Information about the payer.	`ebl:PayerInfoType`	See Table 5.6 on page 68.
Custom	A free-form field for your own use, as set by you in the `Custom` element of `SetExpressCheckout` request. Character length and limitations: 256 single-byte alphanumeric characters	`xs:string`	See description.
InvoiceID	Your own invoice or tracking number, as set by you in the element of the same name in `SetExpressCheckout` request . Character length and limitations: 127 single-byte alphanumeric characters	`xs:string`	See description.

TABLE 5.5 GetExpressCheckoutDetailsResponse Fields

Field	Description	Data Type	Possible Values
ContactPhone	Payer's contact telephone number. **NOTE:** PayPal returns a contact telephone number only if your Merchant account profile settings require that the buyer enter one. Character length and limitations: Field mask is XXX-XXX-XXXX (for US numbers) or +XXX XXXXXXX (for international numbers)	xs:string	See description.

NOTE: Empty elements are returned if there is no representative data for the customer. For example, if the customer does not represent a business (and therefore has no business name on file with PayPal), the returned element has no value, like this: `<PayerBusiness/>`.

TABLE 5.6 Response: PayerInfoType Elements

Field	Description	DataType	Possible Values
Payer	Email address of payer. Character length and limitations: 127 single-byte characters	ebl:EmailAddress Type	See description.
PayerID	Unique PayPal customer account identification number. Character length and limitations:13 single-byte alphanumeric characters.	ebl:UserIDType	See description.
PayerStatus	Status of payer. Character length and limitations: 10 single-byte alphabetic characters.	ebl:PayPalUser StatusCodeType	verified unverified
PayerName	First and last name of payer	ebl:PersonName Type	See Table 5.7 on page 69.
PayerCountry	Payer's country of residence in the form of ISO standard 3166 two-character country codes. Character length and limitations: Two single-byte characters	ebl:CountryCode Type	See Appendix F, "Country Codes."
PayerBusiness	Payer's business name. Character length and limitations: 127 single-byte characters	xs:string	See description.
Address	Payer's shipping address information.	xs:string	See Table 5.8 on page 69.

TABLE 5.7 Response: PayerName Elements

Field	Description	DataType	Possible Values
Salutation	Payer's salutation. Character length and limitations: 20 single-byte characters	`xs:string`	See description.
FirstName	Payer's first name. Character length and limitations: 25 single-byte characters	`ebl:PersonName Type`	See description.
MiddleName	Payer's middle name. Character length and limitations: 25 single-byte characters	`ebl:NameType`	See description.
LastName	Payer's last name. Character length and limitations: 25 single-byte characters	`ebl:NameType`	See description.
Suffix	Payer's suffix. Character length and limitations: 12 single-byte characters	`ebl:SuffixType`	See description.

TABLE 5.8 Response: AddressType Fields

Field	Descriptions	Data Type	Possible Values
AddressStat us	Status of street address on file with PayPal	ebl:AddressStatus TypeCode	none Confirmed Unconfirmed
Name	Person's name associated with this address. Character length and limitations: 32 single-byte characters	`xs:string`	See description.
Street1	First street address. Character length and limitations: 100 single-byte characters	`xs:string`	See description.
Street2	Second street address. Character length and limitations: 100 single-byte characters	`xs:string`	See description.
CityName	Name of city. Character length and limitations: 40 single-byte characters	`xs:string`	See description.
StateOr Province	State or province Character length and limitations: 40 single-byte characters See Appendix E, "State and Province Abbreviations."	`xs:string`	See description.

***TABLE* 5.8 Response: AddressType Fields**

Field	Descriptions	Data Type	Possible Values
PostalCode	U.S. Zip code or other country-specific postal code. Character length and limitations: 20 single-byte characters	xs:string	See description.
Country	Country code. Character limit: Two single-byte characters.	ebl:CountryCod e Type xs:string	See Appendix F, "Country Codes."

DoExpressCheckoutPayment

With DoExpressCheckoutPayment, you either obtain payment through Express Checkout for a final sale or request authorization for later capture of payment.

Diagram of DoExpressCheckoutPayment Types

The following diagram represents the composition of the
DoExpressCheckoutPaymentRequestType and
DoExpressCheckoutPaymentResponseType elements. Elements required in the request are indicated with an asterisk.

FIGURE 5.3 **DoExpressCheckoutPayment Types**
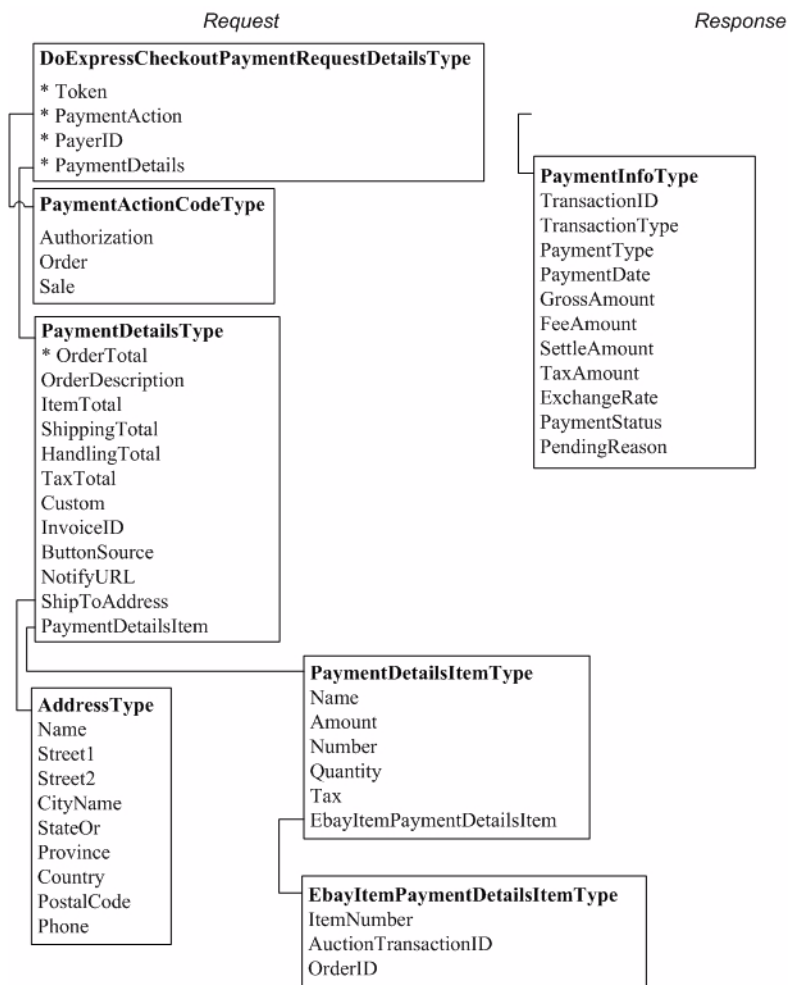
DoExpressCheckoutPaymentRequest

Request to obtain payment with PayPal Express Checkout.

IMPORTANT: PayPal requires that a merchant using Express Checkout display to the customer the same exact amount that the merchant sends to PayPal in the `OrderTotal` element with the `DoExpressCheckoutPaymentRequest` API.

Fields

TABLE 5.9 DoExpressCheckoutPaymentRequest Fields

Field	Description	Data Type	Allowable Values	Required?
Token	The timestamped token value that was returned by `SetExpressCheckout` response and passed on `GetExpressCheckoutDetails` request. Character length and limitations: 20 single-byte characters	`xs:string`	See description.	Yes

PaymentAction	How you want to obtain payment:
	• `Authorization` indicates that this payment is a basic authorization subject to settlement with PayPal Authorization & Capture.
	• `Order` indicates that this payment is is an order authorization subject to settlement with PayPal Authorization & Capture.
	• `Sale` indicates that this is a final sale for which you are requesting payment.
	
	IMPORTANT: You cannot set this value to `Sale` on `SetExpressCheckout` request and then change this value to `Authorization` on the final PayPal Express Checkout API `DoExpressCheckoutPayment` request.
	
	Character length and limit: Up to 13 single-byte alphabetic characters
		`ebl:PaymentActionCodeType`
		`xs:token`
			Default: The transaction resulting from `DoExpressCheckoutPaymentRequest` will be a final sale.
			`Authorization`
			`Order`
			`Sale`
				Yes

PayerID	Unique PayPal customer account identification number as returned by `GetExpressCheckoutDetails` response. Character length and limitations: 13 single-byte alphanumeric characters.	`ebl:UserIDType`	See description.	Yes
PaymentDetails	Information about the payment	`ebl:PaymentDetailsType`	See Table 5.10 on page 73.	Yes

TABLE 5.10 Request: PaymentDetailsType Fields

Field	Description	Data Type	Allowable Values	Required ?
OrderTotal	Total of order, including shipping, handling, and tax. **NOTE:** Limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). **NOTE:** You must set the currencyID attribute to one of the three-character currency codes for any of the supported PayPal currencies.	ebl:BasicAmount Type	See description.	Yes
ItemTotal	Sum of cost of all items in this order. Limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,).	ebl:BasicAmount Type	See description.	No
Shipping Total	Total shipping costs for this order. **NOTE:** Character length and limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). Equivalent to nine characters maximum for USD. **NOTE:** You must set the currencyID attribute to one of the three-character currency codes for any of the supported PayPal currencies.	ebl:BasicAmount Type	See description.	No
Handling Total	Total handling costs for this order. **NOTE:** Character length and limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). Equivalent to nine characters maximum for USD. **NOTE:** You must set the currencyID attribute to one of the three-character currency codes for any of the supported PayPal currencies.	ebl:BasicAmount Type	See description.	No

TABLE 5.10 Request: PaymentDetailsType Fields

Field	Description	Data Type	Allowable Values	Required ?
TaxTotal	Sum of tax for all items in this order. **NOTE:** Character length and limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). Equivalent to nine characters maximum for USD. **NOTE:** You must set the currencyID attribute to one of the three-character currency codes for any of the supported PayPal currencies. Currency code must be set the same as for `OrderTotal`.	`ebl:BasicAmountType`	See description.	No
Order Description	Description of items the customer is purchasing. Character length and limitations: 127 single-byte alphanumeric characters	`xs:string`	See description.	No
Custom	A free-form field for your own use. Character length and limitations: 256 single-byte alphanumeric characters	`xs:string`	See description.	No
InvoiceID	Your own invoice or tracking number. Character length and limitations: 127 single-byte alphanumeric characters	`xs:string`	See description.	No
ButtonSource	An identification code for use by third-party applications to identify transactions. Character length and limitations: 32 single-byte alphanumeric characters	`xs:string`	See description.	No
NotifyURL	Your URL for receiving Instant Payment Notification (IPN) about this transaction. **NOTE:** If you do not specify this value in the request, the notification URL from your Merchant Profile is used, if one exists. Character length and limitations: 2,048 single-byte alphanumeric characters	`xs:string`	See description.	No
ShipToAddress	Address the order will be shipped to.	`ebl:AddressType`	See Table 5.11 on page 75.	No
PaymentDetails Item	Details about each individual item included in the order	`ebl:PaymentDetailsItemType`	See Table 5.12 on page 76.	No

NOTE: The AddressType elements are required only if you include the Address element in the SOAP request.

IMPORTANT: In the following table, any fields in the WSDL or XSD files that are not described here are ignored. For example, do not set CountryName.

TABLE 5.11 Request: ShipToAddress: AddressType Fields

Field	Descriptions	Data Type	Allowable Values	Required?
Name	Person's name associated with this address. Character length and limitations: 32 single-byte characters	xs:string	See description.	Yes
Street1	First street address. Character length and limitations: 100 single-byte characters	xs:string	See description.	Yes
Street2	Second street address. Character length and limitations: 100 single-byte characters	xs:string	See description.	No
CityName	Name of city. Character length and limitations: 40 single-byte characters	xs:string	See description.	Yes
StateOr Province	State or province. Character length and limitations: 40 single-byte characters Required for US addresses only.	xs:string	See Appendix E, "State and Province Abbreviations."	No
PostalCode	U.S. ZIP code or other country-specific postal code. Character length and limitations: 20 single-byte characters	xs:string	See description.	Yes
Country	Country code. Character limit: Two single-byte characters	ebl:CountryCode Type xs:string	See Appendix F, "Country Codes."	Yes
Phone	Phone number. Character length and limit: 20 single-byte characters	xs:string	See description.	No

TABLE 5.12 Request: PaymentDetailsItemType Fields

Field	Description	Data Type	Allowable Values	Required ?
Name	Item name. Character length and limitations: 127 single-byte characters	xs:string	See description.	No
Amount	Cost of item **NOTE:** Character length and limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). Equivalent to nine characters maximum for USD. **NOTE:** You must set the currencyID attribute to one of the three-character currency codes for any of the supported PayPal currencies.	ebl:BasicAmount Type	See description.	No
Number	Item number. Character length and limitations: 127 single-byte characters	xs:string	See description.	No
Quantity	Item quantity. Character length and limitations: Any positive integer	xs:string	See description.	No
Tax	Item sales tax. **NOTE:** Character length and limitations: Must not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator must be a period (.), and the optional thousands separator must be a comma (,). Equivalent to nine characters maximum for USD. **NOTE:** You must set the currencyID attribute to one of the three-character currency codes for any of the supported PayPal currencies.	ebl:BasicAmount Type	See description.	No
EbayItem Payment DetailsItem	Information relating to an auction sale on eBay	eBl:ebayItem PaymentDetails ItemType	See Table 5.13 on page 77	No

If you pass the EbayItemPaymentDetailsItem file, you must set ChannelType to eBayItem and PaymentAction to Sale.

TABLE 5.13 Request: EbayItemPaymentDetailsItemType Fields

Field	Description	Data Type	Allowable Values	Required?
ItemNumber	Auction item number Character length: 765 single-byte characters	xs:string	See description.	No
Auction Transaction ID	Auction transaction identification number Character length: 255 single-byte characters	xs:string	See description.	No
OrderID	Auction order identification number Character length: 64 single-byte characters	xs:string	See description.	No

DoExpressCheckoutPaymentResponse

Response to DoExpressCheckoutPaymentRequest.

Fields

TABLE 5.14 DoExpressCheckoutPaymentResponse Fields

Field	Description	Data Type	Possible Values
Token	The timestamped token value that was returned by SetExpressCheckout response and passed on GetExpressCheckoutDetails request. Character length and limitations:20 single-byte characters	xs:string	See the description of Table 5.3, "SetExpressCheckoutResponse Fields."
PaymentInfo	Information about the payment.	ebl:PaymentInfo Type	See Table 5.15 on page 77.

TABLE 5.15 Response: PaymentInfoType Elements

Element	Description	Data Type	Possible Values
Transaction ID	Unique transaction ID of the payment. **NOTE:** If the PaymentAction of the request was Authorization or Order, this value is your AuthorizationID for use with the Authorization & Capture APIs. Character length and limitations:19 single-byte characters	xs:string	Transaction-specific

TABLE 5.15 Response: PaymentInfoType Elements

Element	Description	Data Type	Possible Values
Transaction Type	The type of transaction Character length and limitations:15 single-byte characters	ns:Payment TransactionCod eType	cart express-checkout
PaymentType	Indicates whether the payment is instant or delayed. Character length and limitations: Seven single-byte characters	ebl:PaymentCod e Type xs:string	none echeck instant
PaymentDate	Time/date stamp of payment	xs:dateTime	Transaction-specific
GrossAmount	The final amount charged, including any shipping and taxes from your Merchant Profile. Character length and limitations: Does not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator is a period (.), and the optional thousands separator is a comma (,). Equivalent to nine characters maximum for USD.	ebl:BasicAmoun t Type	Transaction-specific
FeeAmount	PayPal fee amount charged for the transaction Character length and limitations: Does not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator is a period (.), and the optional thousands separator is a comma (,). Equivalent to nine characters maximum for USD.	ebl:BasicAmoun t Type	Transaction-specific
SettleAmoun t	Amount deposited in your PayPal account after a currency conversion.	ebl:BasicAmoun t Type	Transaction-specific
TaxAmount	Tax charged on the transaction. Character length and limitations: Does not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator is a period (.), and the optional thousands separator is a comma (,). Equivalent to nine characters maximum for USD.	ebl:BasicAmoun t Type xs:string	Transaction-specific

TABLE 5.15 Response: PaymentInfoType Elements

Element	Description	Data Type	Possible Values
ExchangeRate	Exchange rate if a currency conversion occurred. Relevant only if your are billing in their non-primary currency. If the customer chooses to pay with a currency other than the non-primary currency, the conversion occurs in the customer's account. Character length and limitations: a decimal that does not exceed 17 characters, including decimal point	xs:string	Transaction-specific
PaymentStatus	Status of the payment: Completed: The payment has been completed, and the funds have been added successfully to your account balance. Pending: The payment is pending. See the PendingReason element for more information.	ebl:PaymentStatusCodeType xs:token	Completed Pending
PendingReason	The reason the payment is pending: • none: No pending reason • address: The payment is pending because your customer did not include a confirmed shipping address and your Payment Receiving Preferences is set such that you want to manually accept or deny each of these payments. To change your preference, go to the Preferences section of your Profile. • echeck: The payment is pending because it was made by an eCheck that has not yet cleared. • intl: The payment is pending because you hold a non-U.S. account and do not have a withdrawal mechanism. You must manually accept or deny this payment from your Account Overview. • multi-currency: You do not have a balance in the currency sent, and you do not have your Payment Receiving Preferences set to automatically convert and accept this payment. You must manually accept or deny this payment. • verify: The payment is pending because you are not yet verified. You must verify your account before you can accept this payment. • other: The payment is pending for a reason other than those listed above. For more information, contact PayPal customer service.	ebl:PendingStatusCodeType xs:token	none address echeck intl multi-currency verify other

TABLE 5.15 Response: PaymentInfoType Elements

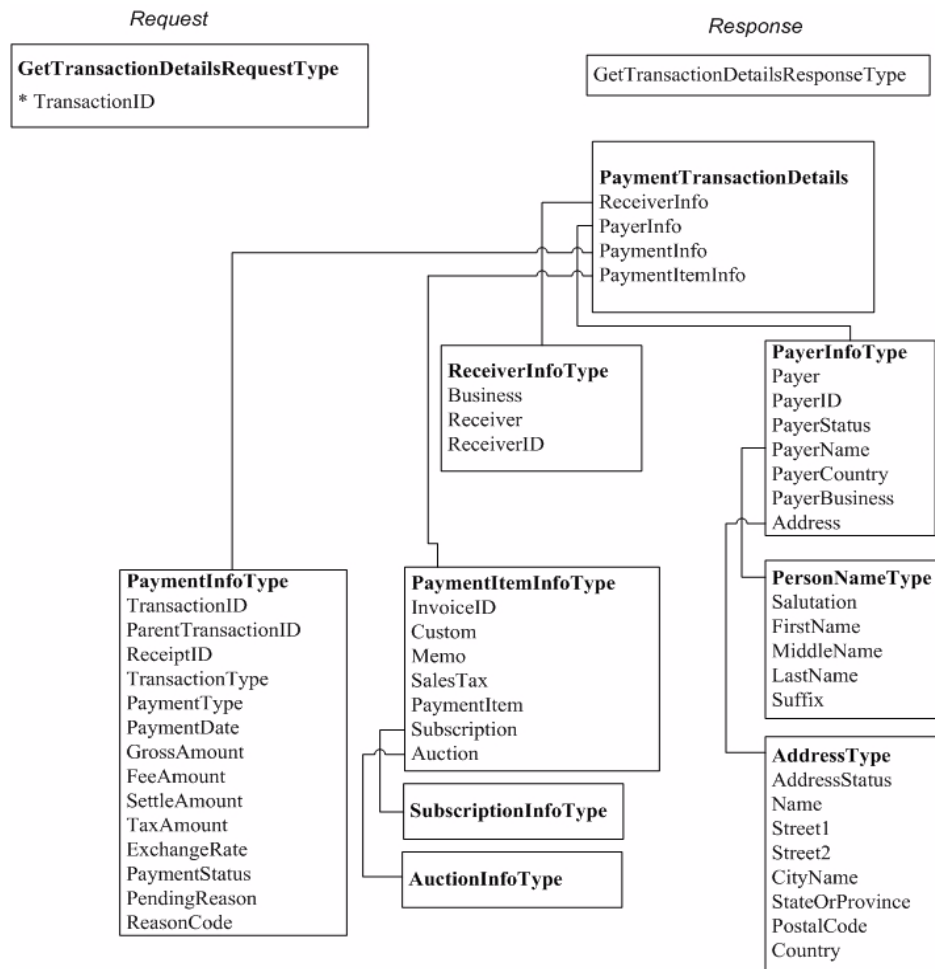| Element	Description	Data Type	Possible Values

ReasonCode	The reason for a reversal if `TransactionType` is reversal:
	• `none`: No reason code
	• `chargeback`: A reversal has occurred on this transaction due to a chargeback by your customer.
	• `guarantee`: A reversal has occurred on this transaction due to your customer triggering a money-back guarantee.
	• `buyer-complaint`: A reversal has occurred on this transaction due to a complaint about the transaction from your customer.
	• `other`: A reversal has occurred on this transaction due to a reason not listed above.
		`ebl:ReasonCode Type`
		`xs:token`
			`none`
			`chargeback`
			`guarantee`
			`buyer-complaint`
			`refund`
			`other`

6 GetTransactionDetails API

The GetTransactionDetails API retrieves information about a single transaction.

Diagram of GetTransactionDetails Types

The following diagram represents the composition of the `GetTransactionDetailsRequestType` and `GetTransactionDetailsResponseType` elements.

FIGURE 6.1 **GetTransactionDetails Types**

GetTransactionDetailsRequest

Get details about a transaction.

Fields

TABLE 6.1 GetTransactionDetailsRequest Fields

Field	Description	Data Type	Possible Values	Required ?
Transaction ID	Unique identifier of a transaction. NOTE: The details for some kinds of transactions cannot be retrieved with GetTransactionDetails. You cannot obtain details of bank transfer withdrawals, for example. Character length and limitations: 17 single-byte alphanumeric characters	xs:string	Transaction-specific	Yes

GetTransactionDetailsResponse

Response to GetTransactionDetailsRequest.

NOTE: Because a field is defined in the formal structure of GetTransactionDetailsResponse does not mean that that field is necessarily returned. Data are returned in a response only if PayPal has recorded data that corresponds to the field.

Fields

For error codes and messages and their meanings, see Appendix A, "Error Codes and Messages for All PayPal APIs."

TABLE 6.2 GetTransactionDetailsResponse Fields

Field	Included Element	Description	Data Type and Possible Values
Payment Transaction Details	ReceiverInfo PayerInfo PaymentInfo PaymentItemInfo	Wrapper structure	
ReceiverInfo		Includes elements described below	ebl:ReceiverInfoType

TABLE 6.2 GetTransactionDetailsResponse Fields

Field	Included Element	Description	Data Type and Possible Values
	Business	Email address or account ID of the payment recipient (the seller). Equivalent to `Receiver` if payment is sent to primary account. Character length and limitations: 127 single-byte alphanumeric characters	`xs:string`
	Receiver	Primary email address of the payment recipient (the seller). If you are the recipient of the payment and the payment is sent to your non-primary email address, the value of Receiver is still your primary email address. Character length and limitations: 127 single-byte alphanumeric characters	`xs:string`
	ReceiverID	Unique account ID of the payment recipient (the seller). This value is the same as the value of the recipient's referral ID.	`xs:string`
PayerInfo		Includes elements described below	`ebl:PayerInfo Type`
	Payer	Email address of payer Character length and limitations: 127 single-byte characters	`ns:EmailAddres s Type`
	PayerID	Unique customer ID. Character length and limitations: 13 single-byte alphanumeric characters.	`ebl:UserIDType`

	PayerStatus	Status of payer's email address:
		`Verified`
		`Unverified`
			`ebl:PayPalUser StatusCodeType`
			`Verified`
			`Unverified`

PayerName (included in PayerInfo)	Salutation	Payer's salutation Character length and limitations: 20 single-byte characters	`xs:string`
	FirstName	Payer's first name Character length and limitations: 25 single-byte characters	`ebl:NameType`
	MiddleName	Payer's middle name Character length and limitations: 25 single-byte characters	`ebl:NameType`

TABLE 6.2 GetTransactionDetailsResponse Fields

Field	Included Element	Description	Data Type and Possible Values
	LastName	Payer's last name Character length and limitations: 25 single-byte characters	`ebl:NameType`
	Suffix	Payer's suffix Character length and limitations: 12 single-byte characters	`ebl:SuffixType`
	PayerCountry	Payment sender's country of residence using standard two-character ISO 3166 country codes. Character length and limitations: Two single-byte characters	`ebl:CountryCodeType`
	PayerBusiness	Payer's business name. Character length and limitations: 127 single-byte characters	`xs:string`
Address (included in `PayerInfo`)		Includes the following elements.	`ebl:AddressType`

	AddressOwner	eBay company that maintains this address	`ebl:AddressOwnerTypeCode`
			`xs:token`
			`eBay`
			`PayPal`

	AddressStatus	Status of the address on file with PayPal:
		`None`
		`Confirmed`
		`Unconfirmed`
			`ebl:AddressStatusCodeType`
			`xs:token`
			`None`
			`Confirmed`
			`Unconfirmed`

	Name	Person's name associated with this address. Character length and limitations: 32 single-byte alphanumeric characters	`xs:string`
	Street1	First street address. Character length and limitations: 300 single-byte alphanumeric characters	`xs:string`
	Street2	Second street address. Character length and limitations: 300 single-byte alphanumeric characters	`xs:string`

TABLE 6.2 GetTransactionDetailsResponse Fields

Field	Included Element	Description	Data Type and Possible Values
	`CityName`	Name of city. Character length and limitations: 120 single-byte alphanumeric characters	`xs:string`
	`StateOrProvince`	State or province. Character length and limitations: 120 single-byte alphanumeric characters See Appendix E, "State and Province Abbreviations."	`xs:string`
	`Country`	Payer's "ship to" country as standard two-character ISO 3166 country code. Character length and limitations: Two single-byte characters	`ns:Country`
	`CountryName`	Expanded name of country. Character length and limitations: 64 single-byte alphanumeric characters	`xs:string`
	`Phone`	Phone number associated with this address	xs:string
	`PostalCode`	Postal code	xs:string
PaymentInfo		Includes the following elements.	`ebl:PaymentInfo Type`
	`ParentTransactionID`	Original transaction to which this transaction is related. This field is populated for the following transaction types: • Reversal • Capture of an authorized transaction. • Reauthorization of a transaction. • Capture of an order. The value of `ParentTransactionID` is the original `OrderID`. • Authorization of an order. The value of `ParentTransactionID` is the original `OrderID`. • Capture of an order authorization. • Void of an order. The value of `ParentTransactionID` is the original `OrderID`. Character length and limitations: 19 single-byte characters	`xs:string`

TABLE 6.2 GetTransactionDetailsResponse Fields

Field	Included Element	Description	Data Type and Possible Values
	TransactionID	PayPal transaction identification number Character length and limitations: 19 single-byte characters	`xs:string`
	ReceiptID	Receipt ID Character length and limitations: 16 digits in `xxxx-xxxx-xxxx-xxxx` format	`xs:string`

	TransactionTyp e	The type of transaction
		`cart`: Transaction created by customer via the PayPal Shopping Cart feature.
		`send-money`: Transaction created by customer from the `Send Money` tab on the PayPal website.
		`web-accept`: Transaction created by customer via Buy Now, Donation, or Auction Smart Logos.
		`subscr-*`: Transaction created by customer via Subscription. `eot` means "end of subscription term."
		`merch-pmt`: preapproved payment.
		`virtual-terminal`: Transaction created via merchant virtual terminal.
			`ebl:Payment TransactionCod e Type`
			`xs:token`
			`cart`
			`send_money`
			`web-accept`
			`subscr-failed`
			`subscr-cancel`
			`subscr-payment`
			`subscr-signup`
			`subscr-eot`
			`subscr-modify`
			`mercht-pmt`
			`mass-pay`
			`virtual-terminal`
			`express-checkout`

	PaymentType	Indicates whether the payment is instant or delayed.
		Character length and limitations: Seven single-byte characters
			`ebl:PaymentCod e Type`
			`xs:string`
			`none`
			`echeck`
			`instant`

	PaymentDate	Date and time of payment	`xs:dateTime`
	GrossAmount	Full amount of the customer's payment, before transaction fee is subtracted	`xs:string`
	FeeAmount	Transaction fee associated with the payment	`xs:string`

TABLE 6.2 GetTransactionDetailsResponse Fields

Field	Included Element	Description	Data Type and Possible Values
	SettleAmount	Amount deposited into the account's primary balance after a currency conversion from automatic conversion through your `Payment Receiving Preferences` or manual conversion through manually accepting a payment. This amount is calculated after fees and taxes have been assessed.	`ebl:BasicAmount Type`
	TaxAmount	Amount of tax for transaction	`xs:string`
	ExchangeRate	Exchange rate for transaction	`xs:string`

	PaymentStatus	Status of the payment. The status of the payment:
		• `None`: No status
		• `Canceled-Reversal`: This means a reversal has been canceled. For example, you won a dispute with the customer, and the funds for the transaction that was reversed have been returned to you.
		• `Completed`: The payment has been completed, and the funds have been added successfully to your account balance.
		• `Denied`: You denied the payment. This happens only if the payment was previously pending because of possible reasons described for the PendingReason element.
		• `Expired`: the authorization period for this payment has been reached.
		• `Failed`: The payment has failed. This happens only if the payment was made from your customer's bank account.
		• `Pending`: The payment is pending. See the `PendingReason` field for more information.
		• `Refunded`: You refunded the payment.
		• `Reversed`: A payment was reversed due to a chargeback or other type of reversal. The funds have been removed from your account balance and returned to the buyer. The reason for the reversal is specified in the `ReasonCode` element.
		• `Processed`: A payment has been accepted.
		• `Voided`: An authorization for this transaction has been voided.
			`ebl:PaymentStatus CodeType`
			 `xs:token`
			 None
			 Canceled-Reversal
			 Completed
			 Denied
			 Expired
			 Failed
			 In-Progress
			 Partially-Refunded
			 Pending
			 Processed
			 Refunded
			 Reversed
			 Voided

TABLE 6.2 GetTransactionDetailsResponse Fields

Field	Included Element	Description	Data Type and Possible Values
	PendingReason	**NOTE:** PendingReason is returned in the response only if PaymentStatus is Pending. The reason the payment is pending: • none: No pending reason • address: The payment is pending because your customer did not include a confirmed shipping address and your Payment Receiving Preferences is set such that you want to manually accept or deny each of these payments. To change your preference, go to the Preferences section of your Profile. • echeck: The payment is pending because it was made by an eCheck that has not yet cleared. • intl: The payment is pending because you hold a non-U.S. account and do not have a withdrawal mechanism. You must manually accept or deny this payment from your Account Overview. • multi-currency: You do not have a balance in the currency sent, and you do not have your Payment Receiving Preferences set to automatically convert and accept this payment. You must manually accept or deny this payment. • verify: The payment is pending because you are not yet verified. You must verify your account before you can accept this payment. • other: The payment is pending for a reason other than those listed above. For more information, contact PayPal Customer Service.	ebl:PendingStatus CodeType xs:token none authorization address echeck intl multi-currency unilateral upgrade verify other

TABLE 6.2 GetTransactionDetailsResponse Fields

Field	Included Element	Description	Data Type and Possible Values

	ReasonCode	The reason for a reversal if `TransactionType` is reversal:
		• `none`: No reason code
		• `chargeback`: A reversal has occurred on this transaction due to a chargeback by your customer.
		• `guarantee`: A reversal has occurred on this transaction due to your customer triggering a money-back guarantee.
		• `buyer-complaint`: A reversal has occurred on this transaction due to a complaint about the transaction from your customer.
		• `other`: A reversal has occurred on this transaction due to a reason not listed above.
			`ebl:ReasonCodeType`
			
			`xs:token`
			`none`
			`chargeback`
			`guarantee`
			`buyer-complaint`
			`refund`
			`other`

PaymentItem Info		Includes the following elements	`ebl:PaymentItemInfoType`
	InvoiceID	Invoice number you set in the original transaction. Character length and limitations: 127 single-byte alphanumeric characters	`xs:string`
	Custom	Custom field you set in the original transaction. Character length and limitations: 127 single-byte alphanumeric characters	`xs:string`
	Memo	Memo entered by your customer in PayPal Website Payments `note` field. Character length and limitations: 255 single-byte alphanumeric characters	`xs:string`
PaymentItem	SaleTax	Amount of tax charged on payment	`xs:string`
	Name	Item name set by you or entered by the customer. If this was a shopping cart transaction, PayPal appends the number of the item to the HTML `item_name` variable. For example, `item_name1`, `item_name2`, and so forth. Character length and limitations: 127 single-byte alphanumeric characters	`xs:string`
	Number	Item number set by you. If this was a shopping cart transaction, PayPal appends the number of the item to the HTML `item_number` variable. For example, `item_number1`, `item_number2`, and so forth. Character length and limitations: 127 single-byte alphanumeric characters	`xs:string`

TABLE 6.2 GetTransactionDetailsResponse Fields

Field	Included Element	Description	Data Type and Possible Values
	Quantity	Quantity set by you or entered by the customer. Character length and limitations: no limit	xs:string
	Amount	Cost of item	ebl:BasicAmount Type
	Options	PayPal item options for shopping cart	ns:OptionType name: xs:string value: xs:string
Subscription		Includes the following elements	ebl:Subscription InfoType
	SubscriptionID	ID generated by PayPal for the subscriber. Character length and limitations: no limit	xs:string
	SubscriptionDate	Subscription start date	xs:dateTime
	EffectiveDate	Date when the subscription modification will be effective	xs:dateTime
	RetryTime	Date PayPal will retry a failed subscription payment.	xs:dateTime
	Username	Username generated by PayPal and given to subscriber to access the subscription. Character length and limitations: 64 alphanumeric single-byte characters	xs:string
	Password	Password generated by PayPal and given to subscriber to access the subscription. For security, the value of the password is hashed. Character length and limitations: 128 alphanumeric single-byte characters	xs:string
	Recurrences	The number of payment installments that will occur at the regular rate. Character length and limitations: no limit	xs:string
	reattempt	Indicates whether reattempts should occur upon payment failures	xs:string

TABLE 6.2 GetTransactionDetailsResponse Fields

Field	Included Element	Description	Data Type and Possible Values
	recurring	Indicates whether regular rate recurs. 1 = Yes	xs:string
Subscription TermsType	Amount	The amount subscriber is to be charged in one payment. Character length and limitations: no limit	eb:BasicAmount Type
	period	The period of time that the subscriber will be charged. Character length and limitations: no limit	xs:string
Auction		Includes the following elements	ebl:AuctionInf o Type
	BuyerID	Customer's auction ID	xs:string
	ClosingDate	Auction's close date	xs:string
	multiItem	 Counter used for multi-item auction payments	xs:string

7 MassPay API

The MassPay API is the programmatic counterpart to the Mass Payment feature available through the PayPal website. You can include up to 250 payments per `MassPayRequest`. You must have a PayPal balance in each currency in which you want to make a payment and have sufficient funds to cover the entire Mass Payment amount and fee. You cannot mix currencies in the same `MassPayRequest`; a separate request is required for each currency.

The posting of payment transactions to individual recipient accounts is not handled by the MassPay API but by a separate PayPal process.

Instant Payment Notification

The MassPay API triggers the sending of an Instant Payment Notification to the recipient of the funds. You must check the IPN to get the necessary information you need about the transaction.

Diagram of MassPay Types

The following diagram represents the composition of the `MassPayRequestType`, `MassPayRequestItemType`, and `MassPayResponseType` elements.

FIGURE 7.1 **MassPay Types**
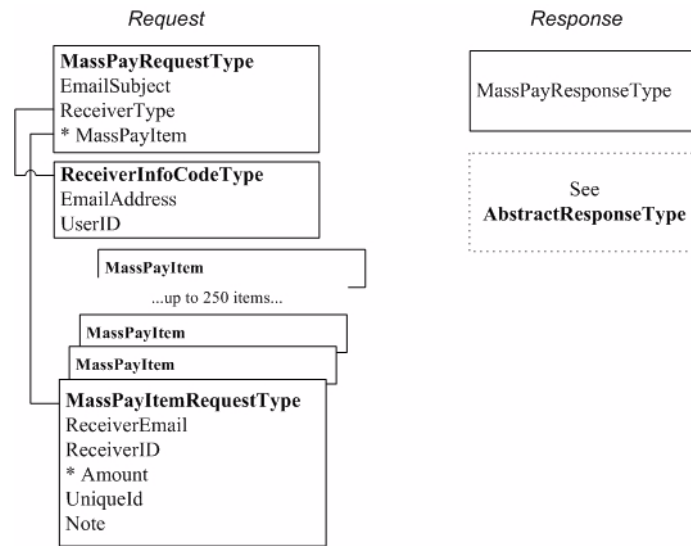
MassPayRequest

Process payments *en masse* to up to 250 different recipients.

Fields

TABLE 7.1 **MassPayRequestType Fields**

Field	Description	Data Type	Possible Values	Required?
EmailSubject	The subject line of the email that PayPal sends when the transaction is completed. The subject line is the same for all recipients. Character length and limitations: 255 single-byte alphanumeric characters	xs:string	See description.	No
ReceiverType	Indicates how you identify the recipients of payments in all `MassPayItems`: either by `EmailAddress` (`ReceiverEmail` in `MassPayItem`) or by `UserID` (`ReceiverID` in `MassPayItem`).	ebl:ReceiverInfo CodeType xs:token	See Table 7.2.	No
MassPayItem	Details of each payment. **NOTE:** A single `MassPayRequest` can include up to 250 `MassPayItems`	ebl:MassPayItem Type	See Table 7.3.	Yes

TABLE 7.2 ReceiverInfoCodeType Enumeration

Field	Description	Required?
EmailAddress	Email address of recipient on file with PayPal. Character length and limitations: 12 single-byte alphanumeric characters.	No
UserID	Unique PayPal customer account number. Character length and limitations: 6 single-byte characters maximum.	No

TABLE 7.3 MassPayItemType Fields

Field	Description	Data Type	Possible Values	Required?
ReceiverEmail	Email address of recipient. **IMPORTANT:** You must specify either `ReceiverEmail` or `ReceiverID`, but you must not mix `ReceiverEmail` and `ReceiverID` in the group of MassPayItems. Use only one or the other, but not both, in a single request. Character length and limitations: 127 single-byte characters maximum.	ebl:EmailAddressType	See description.	No
ReceiverID	Unique PayPal customer account number. This value corresponds to the value of `PayerID` returned by `GetTransactionDetails`. **IMPORTANT:** You must specify either `ReceiverEmail` or `ReceiverID`, but you must not mix `ReceiverEmail` and `ReceiverID` in the group of MassPayItems. Use only one or the other, but not both, in a single request. Character length and limitations: 17 single-byte characters maximum.	xs:string	Transaction-specific	No

TABLE 7.3 MassPayItemType Fields

Field	Description	Data Type	Possible Values	Required ?
Amount	Payment amount. **NOTE:** You must set the `currencyID` attribute to one of the three-character currency codes for any of the supported PayPal currencies. See "" on page 23. **IMPORTANT:** You cannot mix currencies in a single MassPayRequest. A single request must include items that are of the same currency.	`ebl:BasicAmount Type`	Transaction-specific	Yes
UniqueId	Transaction-specific identification number for tracking in an accounting system. Character length and limitations: 30 single-byte characters. No whitespace allowed.	`xs:string`	Transaction-specific	No
Note	Custom note for each recipient. Character length and limitations: 4,000 single-byte alphanumeric characters	`xs:string`	Transaction-specific	No

MassPayResponse

Response to `MassPayRequest`.

Fields

The elements returned are the same as for `AbstractResponseType`. See "AbstractResponseType" on page 28.

For error codes and messages and their meanings, see Appendix A, "Error Codes and Messages for All PayPal APIs."

8 RefundTransaction API

Use the RefundTransaction API to make full or partial refunds.

Diagram of RefundTransaction Types

The following diagram represents the composition of the
`RefundTransactionRequestType` and `RefundTransactionResponseType` elements.

FIGURE 8.1 RefundTransaction Types
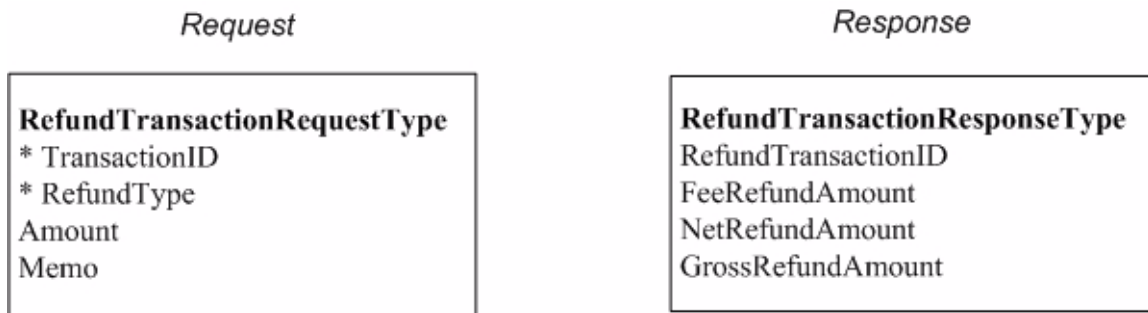
RefundTransactionRequest

Request to refund a payment.

With `RefundTransactionRequest`, you must always pass the values of `TransactionID`
and `RefundType`. If `RefundType` is set to `Full`, you must *not* set Amount; however,

if `RefundType` is `Partial`, you *must* set `Amount`.

Fields

TABLE 8.1 RefundTransactionRequest Fields

Field	Description	Data Type	Possible Values	Required ?
TransactionID	Unique identifier of a transaction Character length and limitations: 17 single-byte alphanumeric characters	`xs:string`	Transaction-specific	Yes

RefundType	Type of refund you are making
	• `Other`
	• `Full`
	• `Partial`
		`ebl:RefundPurp oseTypeCodeTyp e`
		`xs:token`
			`Other`
			`Full`
			`Partial`
				Yes

Amount	Refund amount. `Amount` is required if `RefundType` is `Partial`. NOTE: If `RefundType` is `Full`, do not set `Amount`.	`ebl:BasicAmoun t Type`	Transaction-specific	See description
Memo	Custom memo about the refund. Character length and limitations: 255 single-byte alphanumeric characters	`xs:string`	Transaction-specific	No

RefundTransactionResponse

Response to `RefundTransactionRequest`.

Fields

TABLE 8.2 RefundTransactionResponse Fields

Field	Description	Data Type	Value
Refund Transaction ID	Unique transaction ID of the refund. Character length and limitations:17 single-byte characters	`xs:string`	See description.
FeeRefund Amount	Transaction fee refunded to original recipient of payment	ebl:BasicAmount Type	See description.
GrossRefund Amount	Amount of money refunded to original payer	ebl:BasicAmount Type	See description.

TABLE 8.2 RefundTransactionResponse Fields

Field	Description	Data Type	Value
NetRefund Amount	Amount subtracted from PayPal balance of original recipient of payment to make this refund	ebl:BasicAmount Type	See description.

9 TransactionSearch API

With `TransactionSearchRequest` you must always set the StartDate field. Some other behavior:

- Setting `TransactionID` overrides all other fields (even the required StartDate field).
- The effect of setting other elements is additive or can alter the search criteria.

`TransactionSearchResponse` returns up to 100 matches. Partial matches are displayed. For example, setting the `TransactionSearchRequest FirstName` to "Jess" returns results such as "Jessica" and "Jesse".

The most important returned element is `TransactionID`, which you can pass to `GetTransactionDetailsRequest` in order to retrieve all available information about a specific transaction.

Diagram of TransactionSearch Types

The following diagram represents the composition of the `TransactionSearchRequestType` and `PaymentTransactionSearchResultsType` elements.

FIGURE 9.1 *TransactionSearch Types*
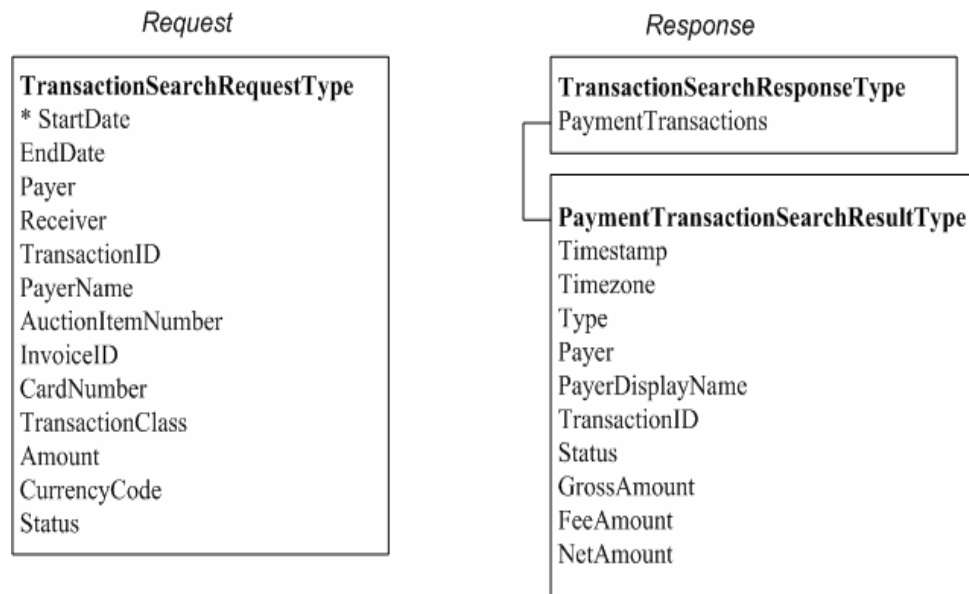
TransactionSearchRequest

Request to search transaction history.

Fields

TABLE 9.1 *TransactionSearchRequest Fields*

Field	Description	Data Type	Allowable Values	Required ?
StartDate	The earliest transaction date at which to start the search. NOTE: No wildcards are allowed. The value must be in UTC/GMT format.	xs:dateTime	Transaction-specific	Yes
EndDate	The latest transaction date to be included in the search	xs:dateTime	Transaction-specific	No
Payer	Search by the buyer's email address Character length and limitations: 127 single-byte alphanumeric characters	ebl:EmailAddress Type xs:string	Transaction-specific	No

TABLE 9.1 TransactionSearchRequest Fields

Field	Description	Data Type	Allowable Values	Required ?
Receiver	Search by the receiver's email address. If the merchant account has only one email, this is the primary email. Can also be a non-primary email.	`ebl:EmailAddress Type` `xs:string`	Transaction-specific	No
ReceiptID	Search by the PayPal Account Optional receipt ID	`xs:string`	Transaction-specific	No
TransactionID	Search by the transaction ID. **NOTE:** The returned results are from the merchant's transaction records. Character length and limitations: 19 single-byte characters maximum	`ebl:TransactionID` `xs:string`	Transaction-specific	No
InvoiceID	Search by invoice identification key, as set by you for the original transaction. This field searches the records for items sold by the merchant, not the items purchased. **NOTE:** No wildcards are allowed. Character length and limitations: 127 single-byte characters maximum	`xs:string`	Transaction-specific	No
CardNumber	Search by credit card number, as set by you for the original transaction. This field searches the records for items sold by the merchant, not the items purchased. **NOTE:** No wildcards are allowed. Character length and limitations: Must be at least 11 and no more than 25 single-byte numeric characters maximum. Special punctuation, such as dashes or spaces, is ignored.	`xs:string`	Transaction-specific	No
PayerName	Search by the buyer's name • Salutation: 20 single-byte character limit • FirstName: 25 single-byte character limit • MiddleName: 25 single-byte character limit • LastName: 25 single-byte character limit • Suffix: 12 single-byte character limit	`ebl:PersonName Type` `xs:string`	Transaction-specific	No
AuctionItem Number	. Search by auction item number of the purchased goods	`xs:string`	Transaction-specific	No

TABLE 9.1 TransactionSearchRequest Fields

Field	Description	Data Type	Allowable Values	Required ?
Transaction Class	Search by classification of transaction. **NOTE:** Some kinds of possible classes of transactions are not searchable with `this field`. You cannot search for bank transfer withdrawals, for example. • `All`: all transaction classifications • `Sent`: only payments sent • `Received`: only payments received • `MoneyRequest`: only money requests • `FundsAdded`: only funds added to balance • `FundsWithdrawn`: only funds withdrawn from balance • `Referral`: only transactions involving referrals • `Fee`: only transactions involving fees • `Subscription`: only transactions involving subscriptions • `Dividend`: only transactions involving dividends • `CurrencyConversions`: only transactions involving currency conversions • `BalanceTransfer`: only transactions involving balance transfers • `Shipping`: only transactions involving UPS shipping fees • `BalanceAffecting`: only transactions that affect the account balance • `ECheck`: only transactions involving eCheck	`ebl:Payment TransactionCla ss CodeType` `xs:token`	See description.	No
Amount	Search by transaction amount **NOTE:** You must set the `currencyID` attribute to one of the three-character currency codes for any of the supported PayPal currencies.	`ebl:BasicAmoun t Type` `xs:string`	Transaction-specific	No
CurrencyCode	Search by currency code	`xs:token`	Transaction-specific	No

TABLE 9.1 TransactionSearchRequest Fields

Field	Description	Data Type	Allowable Values	Required ?

Status	Search by transaction status:
	• `Pending`: The payment is pending. The specific reason the payment is pending is returned by the GetTransactionDetails API `PendingReason` field.
	• `Processing`: The payment is being processed.
	• `Success`: The payment has been completed and the funds have been added successfully to your account balance.
	• `Denied`: You denied the payment. This happens only if the payment was previously pending.
	• `Reversed`: A payment was reversed due to a chargeback or other type of reversal. The funds have been removed from your account balance and returned to the buyer.
		`ebl:Payment Transaction StatusCodeType`
		`xs:token`
			`Pending`
			`Processing`
			`Success`
			`Denied`
			`Reversed`
				No

TransactionSearchResponse

Response to `TransactionSearchRequest`.

Fields

For error codes and messages and their meanings, see Appendix A, "Error Codes and Messages for All PayPal APIs."

TABLE 9.2 PaymentTransactions Fields

Field	Description	Data Type	Possible Values
Timestamp	The date and time (in UTC/GMT format) the transaction occurred	`xs:dateTime`	Transaction-specific
Timezone	The time zone of the transaction	`xs:string`	Transaction-specific
Type	The type of the transaction	`xs:string`	Transaction-specific

TABLE 9.2 PaymentTransactions Fields

Field	Description	Data Type	Possible Values

Payer	The email address of either the payer or the payment recipient (the "payee"). If the payment amount is positive, this field is the recipient of the funds. If the payment is negative, this field is the paying customer.	`ebl:EmailAddre`
		`ss`
		`Type`
		`xs:string`
			Transaction-specific

Payer DisplayName	Display name of the payer	`xs:string`	Transaction-specific
TransactionID	Seller's transaction ID	`xs:string`	Transaction-specific
Status	The status of the transaction.	`xs:string`	Transaction-specific

GrossAmount	The total gross amount charged, including any profile shipping cost and taxes	`ebl:BasicAmoun`
		`t`
		`Type`
		`xs:string`
			Transaction-specific

FeeAmount	The fee that PayPal charged for the transaction	`BasicAmount`
		`Type`
		`xs:string`
			Transaction-specific

NetAmount	The net amount of the transaction	`BasicAmount`
		`Type`
		`xs:string`
			Transaction-specific

A   Error Codes and Messages for All PayPal APIs

General API Errors

TABLE *A.1 General API Errors*

Error Code	Short Message	Long Message	Correcting This Error
10002	Authentication /Authorization Failed	Username/Password is incorrect	This error can be caused by an incorrect API username, an incorrect API password, or an invalid API signature. Make sure that all three of these values are correct. For your security, PayPal does not report exactly which of these three values might be in error.
10002	Authentication /Authorization Failed	You do not have permissions to make this API call	
10002	Authentication /Authorization Failed	Account is locked or inactive	
10002	Internal Error	Internal Error	
10002	Authentication /Authorization Failed	Internal Error	
10002	Authentication /Authorization Failed	Account is not verified	
10002	Authentication /Authorization Failed	This call is not defined in the database!	
10002	Authentication /Authorization Failed	Token is not valid	

TABLE A.1 General API Errors

Error Code	Short Message	Long Message	Correcting This Error
10002	Restricted account	Account is restricted	Your PayPal merchant account has been restricted. Contact your PayPal account manager for resolution.
10002	Authentication /Authorization Failed	Token is not valid	
10002	Authentication /Authorization Failed	API access is disabled for this account	
10002	Authentication /Authorization Failed	Client certificate is disabled	
10002	Restricted account	Account is restricted	

Direct Payment API Errors

TABLE A.2 Direct Payment API Errors

Error Code	Short Message	Long Message	Corrective Action
10500	Invalid Configuration	This transaction cannot be processed due to an invalid merchant configuration.	Occurs when you have not agreed to the billing agreement.
10501	Invalid Configuration	This transaction cannot be processed due to an invalid merchant configuration.	Occurs when the billing agreement is disabled or inactive.
10502	Invalid Data	This transaction cannot be processed. Please use a valid credit card.	The credit card used is expired.
10504	Invalid Data	This transaction cannot be processed. Please enter a valid Credit Card Verification Number.	The CVV provided is invalid. The CVV is between 3-4 digits long.

TABLE *A.2 Direct Payment API Errors*

Error Code	Short Message	Long Message	Corrective Action
10505	Gateway Decline	This transaction cannot be processed.	The transaction was refused because the AVS response returned the value of N, and the merchant account is not able to accept such transactions.
10507	Invalid Configuration	This transaction cannot be processed. Please contact PayPal Customer Service.	Your PayPal account is restricted - contact PayPal for more information.
10508	Invalid Data	This transaction cannot be processed. Please enter a valid credit card expiration date.	The expiration date must be a two-digit month and four-digit year.
10509	Invalid Data	This transaction cannot be processed.	You must submit an IP address of the buyer with each API call.
10510	Invalid Data	The credit card type is not supported. Try another card type.	The credit card type entered is not currently supported by PayPal.
10511	Invalid Data	This transaction cannot be processed.	The merchant selected an value for the PaymentAction field that is not supported.
10512	Invalid Data	This transaction cannot be processed. Please enter a first name.	The first name of the buyer is required for this merchant.
10513	Invalid Data	This transaction cannot be processed. Please enter a last name.	The last name of the buyer is required for this merchant.
10519	Invalid Data	Please enter a credit card.	The credit card field was blank.
10520	Invalid Data	This transaction cannot be processed.	The total amount and item amounts do not match.
10521	Invalid Data	This transaction cannot be processed. Please enter a valid credit card.	The credit card entered is invalid.
10523	Internal Error	This transaction cannot be processed.	None - this is a PayPal internal error.
10525	Invalid Data	This transaction cannot be processed. The amount to be charged is zero.	The merchant entered a amount of zero.
10526	Invalid Data	This transaction cannot be processed. The currency is not supported at this time.	The currency code entered is not supported.
10527	Invalid Data	This transaction cannot be processed. Please enter a valid credit card number and type.	The credit card entered is invalid.

TABLE A.2 Direct Payment API Errors

Error Code	Short Message	Long Message	Corrective Action
10534	Gateway Decline	This transaction cannot be processed. Please enter a valid credit card number and type.	The credit card entered is currently restricted by PayPal. Contact PayPal for more information.
10535	Gateway Decline	This transaction cannot be processed. Please enter a valid credit card number and type.	The credit card entered is invalid.
10536	Invalid Data	This transaction cannot be processed.	The merchant entered an invoice ID that is already associated with a transaction by the same merchant. By default, the invoice ID must be unique for all transactions. To change this setting, log into PayPal or contact customer service.
10537	Filter Decline	This transaction cannot be processed.	The transaction was declined by the country filter managed by the merchant. To accept this transaction, change your risk settings on PayPal.
10538	Filter Decline	This transaction cannot be processed.	The transaction was declined by the maximum amount filter managed by the merchant. To accept this transaction, change your risk settings on PayPal.
10539	Filter Decline	This transaction cannot be processed.	The transaction was declined by PayPal. Contact PayPal for more information.
10540	Invalid Data	The transaction cannot be processed due to an invalid address.	The transaction was declined by PayPal because of an invalid address.
10541	Gateway Decline	This transaction cannot be processed. Please enter a valid credit card number and type.	The credit card entered is currently restricted by PayPal. Contact PayPal for more information.
10542	Invalid Data	This transaction cannot be processed. Please enter a valid email address.	The email address provided by the buyer is in an invalid format.
10544	Gateway Decline	This transaction cannot be processed.	The transaction was declined by PayPal. Contact PayPal for more information.
10545	Gateway Decline	This transaction cannot be processed.	The transaction was declined by PayPal because of possible fraudulent activity. Contact PayPal for more information.
10546	Gateway Decline	This transaction cannot be processed.	The transaction was declined by PayPal because of possible fraudulent activity on the IP address. Contact PayPal for more information.
10547	Internal Error	This transaction cannot be processed.	None - this is a PayPal internal error.

<table>
</table>

<center>TABLE <small>A</small>.2 Direct Payment API Errors</center>

Error Code	Short Message	Long Message	Corrective Action
10548	Invalid Configuration	This transaction cannot be processed. The merchant's account is not able to process transactions.	The merchant account attempting the transaction is not a business account at PayPal. Check your account settings.
10549	Invalid Configuration	This transaction cannot be processed. The merchant's account is not able to process transactions.	The merchant account attempting the transaction is not able to process Direct Payment transactions. Contact PayPal for more information.
10550	Invalid Configuration	This transaction cannot be processed.	Access to Direct Payment was disabled for your account. Contact PayPal for more information.
10552	Invalid Configuration	This transaction cannot be processed.	The merchant account attempting the transaction does not have a confirmed email address with PayPal. Check your account settings.
10553	Gateway Decline	This transaction cannot be processed.	The merchant attempted a transaction where the amount exceeded the upper limit for that merchant.
10554	Filter Decline	This transaction cannot be processed.	The transaction was declined because of a merchant risk filter for AVS. Specifically, the merchant has set to decline transaction when the AVS returned a no match (AVS = N).
10555	Filter Decline	This transaction cannot be processed.	The transaction was declined because of a merchant risk filter for AVS. Specifically, the merchant has set to decline transaction when the AVS returned a partial match.
10556	Filter Decline	This transaction cannot be processed.	The transaction was declined because of a merchant risk filter for AVS. Specifically, the merchant has set to decline transaction when the AVS was unsupported.
10561	Invalid Data	There's an error with this transaction. Please enter complete billing address.	
10562	Invalid Data	This transaction cannot be processed. Please enter a valid credit card expiration year.	
10563	Invalid Data	This transaction cannot be processed. Please enter a valid credit card expiration month.	
10564	Gateway Decline	This transaction cannot be processed.	There was a problem processing this transaction.

TABLE A.2 Direct Payment API Errors

Error Code	Short Message	Long Message	Corrective Action
10565	Merchant country unsupported	The merchant country is not supported.	
10566	Credit card type unsupported	The credit card type is not supported.	
10567	Invalid Data	This transaction cannot be processed. Please enter a valid credit card number and type.	
10701	Invalid Data	There's an error with this transaction. Please enter a valid billing address.	
10702	Invalid Data	There's an error with this transaction. Please enter a valid address1 in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10703	Invalid Data	There's an error with this transaction. Please enter a valid address2 in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10704	Invalid Data	There's an error with this transaction. Please enter a valid city in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10705	Invalid Data	There's an error with this transaction. Please enter a valid state in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10706	Invalid Data	There's an error with this transaction. Please enter a valid postal code in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10707	Invalid Data	There's an error with this transaction. Please enter a valid country in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10708	Invalid Data	There's an error with this transaction. Please enter a complete billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10709	Invalid Data	There's an error with this transaction. Please enter an address1 in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10709	Invalid Data	There's an error with this transaction. Please enter an address1 in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.

TABLE ***A.2 Direct Payment API Errors***

Error Code	Short Message	Long Message	Corrective Action
10710	Invalid Data	There's an error with this transaction. Please enter a city in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10710	Invalid Data	There's an error with this transaction. Please enter a city in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10711	Invalid Data	There's an error with this transaction. Please enter your state in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10712	Invalid Data	There's an error with this transaction. Please enter your five digit postal code in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10713	Invalid Data	There's an error with this transaction. Please enter a country in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10713	Invalid Data	There's an error with this transaction. Please enter a country in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10714	Invalid Data	There's an error with this transaction. Please enter a valid billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10715	Invalid Data	There's an error with this transaction. Please enter a valid state in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10716	Invalid Data	There's an error with this transaction. Please enter your five digit postal code in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10717	Invalid Data	There's an error with this transaction. Please enter a valid postal code in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10718	Invalid Data	There's an error with this transaction. Please enter a valid city and state in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10719	Invalid Data	There's an error with this transaction. Please enter a valid shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.

TABLE A.2 *Direct Payment API Errors*

Error Code	Short Message	Long Message	Corrective Action
10720	Invalid Data	There's an error with this transaction. Please enter a valid address1 in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10721	Invalid Data	There's an error with this transaction. Please enter a valid address2 in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10722	Invalid Data	There's an error with this transaction. Please enter a valid city in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10723	Invalid Data	There's an error with this transaction. Please enter a valid state in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10724	Invalid Data	There's an error with this transaction. Please enter your five digit postal code in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10725	Invalid Data	There's an error with this transaction. Please enter a valid country in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10726	Invalid Data	There's an error with this transaction. Please enter a complete shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10726	Invalid Data	There's an error with this transaction. Please enter a complete shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10727	Invalid Data	There's an error with this transaction. Please enter an address1 in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10727	Invalid Data	There's an error with this transaction. Please enter an address1 in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10728	Invalid Data	There's an error with this transaction. Please enter a city in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10728	Invalid Data	There's an error with this transaction. Please enter a city in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.

TABLE A.2 **Direct Payment API Errors**

Error Code	Short Message	Long Message	Corrective Action
10729	Invalid Data	There's an error with this transaction. Please enter your state in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10730	Invalid Data	There's an error with this transaction. Please enter your five digit postal code in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10731	Invalid Data	There's an error with this transaction. Please enter a country in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10731	Invalid Data	There's an error with this transaction. Please enter a country in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10732	Invalid Data	There's an error with this transaction. Please enter a valid shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10733	Invalid Data	There's an error with this transaction. Please enter a valid state in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10734	Invalid Data	There's an error with this transaction. Please enter your five digit postal code in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10735	Invalid Data	There's an error with this transaction. Please enter your five digit postal code in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10736	Invalid Data	There's an error with this transaction. Please enter a valid city and state in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10744	Invalid Data	This transaction cannot be processed. Please enter a valid country code in the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10745	Invalid Data	This transaction cannot be processed. Please enter a valid country code in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.

TABLE A.2 Direct Payment API Errors

Error Code	Short Message	Long Message	Corrective Action
10746	Invalid Data	This transaction cannot be processed. Please use a valid country on the billing address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10747	Invalid Data	This transaction cannot be processed.	The merchant entered an IP address that was in an invalid format. The IP address must be in a format such as 123.456.123.456.
10748	Invalid Data	This transaction cannot be processed without a Credit Card Verification Number.	The merchant's configuration requires a CVV to be entered, but no CVV was provided with this transaction. Contact PayPal if you wish to change this setting.
10750	Invalid Data	There's an error with this transaction. Please enter a valid state in the shipping address.	There was a problem with a particular field in the address. The long error message will tell you what field is invalid.
10751	Invalid Data	There's an error with this transaction. Please enter a valid state in the billing address.	The merchant provided an address either in the United States or Canada, but the state provided is not a valid state in either country.
10752	Gateway Decline	This transaction cannot be processed.	The transaction was declined by the issuing bank, not PayPal. The merchant should attempt another card.
10754	Gateway Decline	This transaction cannot be processed.	The transaction was declined by PayPal. Contact PayPal for more information.
10755	Invalid Data	This transaction cannot be processed due to an unsupported currency.	The currency code entered by the merchant is not supported.
10756	Gateway Decline	The transaction cannot be processed. The country and billing address associated with this credit card do not match.	None - this is a PayPal internal error.
10758	Invalid Configuration	There's been an error due to invalid API username and/or password.	The API username or password is incorrect for this merchant.
10759	Gateway Decline	This transaction cannot be processed. Please enter a valid credit card number and type.	The transaction was declined by PayPal. Contact PayPal for more information.
10760	Invalid Configuration	This transaction cannot be processed. The country listed for your business address is not currently supported.	The merchant's country of residence listed in their PayPal account is not currently supported to allow Direct Payment transactions.

***TABLE* A.2 Direct Payment API Errors**

Error Code	Short Message	Long Message	Corrective Action
10761	Gateway Decline	This transaction cannot be processed. Please check the status of your first transaction before placing another order.	The transaction was declined because PayPal is currently processing a transaction by the same buyer for the same amount. Can occur when a buyer submits multiple, identical transactions in quick succession.
10762	Gateway Decline	This transaction cannot be processed.	The CVV provide is invalid. The CVV is between 3-4 digits long.
10763	Invalid Data	This transaction cannot be processed.	None - this is a PayPal internal error.
15001	Gateway Decline	This transaction cannot be processed.	The transaction was rejected by PayPal because of excessive failures over a short period of time for this credit card. Contact PayPal for more information.
15002	Gateway Decline	This transaction cannot be processed.	The transaction was declined by PayPal. Contact PayPal for more information.
15003	Invalid Configuration	This transaction cannot be processed.	The transaction was declined because the merchant does not have a valid commercial entity agreement on file with PayPal. Contact PayPal for more information.
15004	Gateway Decline	This transaction cannot be processed. Please enter a valid Credit Card Verification Number.	The transaction was declined because the CVV entered does not match the credit card.
15005	Processor Decline	This transaction cannot be processed.	The transaction was declined by the issuing bank, not PayPal. The merchant should attempt another card.
15006	Processor Decline	This transaction cannot be processed. Please enter a valid credit card number and type.	The transaction was declined by the issuing bank, not PayPal. The merchant should attempt another card.
15007	Processor Decline	This transaction cannot be processed. Please use a valid credit card.	The transaction was declined by the issuing bank because of an expired credit card. The merchant should attempt another card.
15008	Invalid Data	This transaction has been completed, but the total of items in the cart did not match the total of all items.	

Recurring Payments API Errors

SetCustomerBillingAgreement Errors

TABLE A.3 SetCustomerBillingAgreement Errors

Error Code	Short Message	Long Message	Additional Information
10004	Transaction refused because of an invalid argument. See additional error messages for details.	Invalid argument; BillingType input field is set to None	
10404	Transaction refused because of an invalid argument. See additional error messages for details.	ReturnURL is missing.	ReturnURL tag has no content
10405	Transaction refused because of an invalid argument. See additional error messages for details.	CancelURL is missing.	CancelURL tag has no content
10407	Transaction refused because of an invalid argument. See additional error messages for details.	Invalid buyer email address (BuyerEmail).	Invalid BuyerEmail (badly formatted or violates SMTP protocol defined email address format) or BuyerEmail is passed as an empty tag.
10436	Transaction refused because of an invalid argument. See additional error messages for details.	PageStyle value exceeds maximum allowable length.	PageStyle tag is too long
10437	Transaction refused because of an invalid argument. See additional error messages for details.	cpp-header-image value exceeds maximum allowable length.	cpp_header_image tag is too long; maximum length is 127
10438	Transaction refused because of an invalid argument. See additional error messages for details.	cpp-header-border-color value exceeds maximum allowable length.	cpp_header_border_color tag is too long; maximum length is 6

118
     April 2007
     SOAP API Developer Reference

TABLE A.3 SetCustomerBillingAgreement Errors

Error Code	Short Message	Long Message	Additional Information
10439	Transaction refused because of an invalid argument. See additional error messages for details.	cpp-header-back-color value exceeds maximum allowable length.	cpp_header_back_color field is too long; maximum length is 6
10440	Transaction refused because of an invalid argument. See additional error messages for details.	cpp-payflow-color value exceeds maximum allowable length.	cpp_payflow_color field is too long. maximum length is 6
10471	Transaction refused because of an invalid argument. See additional error messages for details.	ReturnURL is invalid.	ReturnURL field contains invalid URL
10472	Transaction refused because of an invalid argument. See additional error messages for details.	CancelURL is invalid.	CancelURL field contains an invalid URL

GetBillingAgreementCustomerDetails Errors

TABLE A.4 GetBillingAgreementCustomerDetails Errors

Error Code	Short Message	Long Message	Additional Information
10408	Missing token	Token is missing	Token is missing
10409	You're not authorized to access this info.	Express Checkout token was issued for a merchant account other than yours.	Token belongs to a different merchant
10410	Invalid token	Invalid token.	Token invalid
10411	This Express Checkout session has expired.	This Express Checkout session has expired. Token value is no longer valid.	Token expired

CreateRecurringPaymentsProfile Errors

TABLE A.5 CreateRecurringPaymentsProfile Errors

Error Code	Short Message	Long Message	Additional Information
11501	Invalid merchant country	Only US is currently supported for merchant country	Merchant country must be US
11502	The token is missing or is invalid	The token is missing or is invalid	Missing token .
11503	Missing subscription details	Missing subscription details	One or more subscription details are missing from the request..
11504	Missing schedule details	Missing schedule details	One or more schedule details are missing from the request.
11505	Start date is required	Subscription start date is required	
11506	Invalid max failed payments	Max failed payments, if supplied, must be >= 0	
11507	Invalid trial billing frequency	Trial billing frequency must be > 0 and be less than or equal to one year	The combination of trial billing period and trial billing frequency cannot exceed one year.
11508	Invalid trial total billing cycles	Trial total billing cycles must be >= 0 (0 means unending)	
11509	Invalid trial billing period	Trial billing period must be one of Day, Week, Month, SemiMonth, or Year	
11510	Invalid trial amount	Trial amount must be >= 0	
11511	Invalid currency for trial amount	USD is the only currently supported currency for trial amount	Currency must be USD.
11512	Invalid trial shipping amount	Trial shipping amount must be >= 0	
11513	Invalid currency for trial amount	USD is the only currently supported currency for trial amount	Currency must be USD.
11514	Trial tax amount must be >= 0	Trial tax amount must be >= 0	
11515	Invalid currency for trial amount	USD is the only currently supported currency for trial amount	Currency must be USD.

TABLE A.5 CreateRecurringPaymentsProfile Errors

Error Code	Short Message	Long Message	Additional Information
11516	Invalid billing frequency	Billing Frequency must be > 0 and be less than or equal to one year	The combination of billing frequency and billing period cannot exceed one year.
11517	Invalid total billing cycles	Total billing cycles must be >= 0 (0 means unending)	
11518	Invalid billing period	Billing period must be one of Day, Week, Month, SemiMonth, or Year	
11519	Invalid amount	Amount must be >= 0	
11520	Invalid currency for trial amount	USD is the only currently supported currency for trial amount	Currency must be USD.
11521	Invalid shipping amount	Shipping amount must be >= 0	
11522	Invalid currency for trial amount	USD is the only currently supported currency for trial amount	Currency must be USD.
11523	Invalid tax amount	Tax amount must be >= 0	
11524	Invalid currency for trial amount	USD is the only currently supported currency for trial amount	Currency must be USD.
11545	Denied	Payer's account is denied	
11546	Denied	Merchant account is denied	

GetTransactionDetails API Errors

TABLE A.6 GetTransactionDetails API Errors

Error Code	Short Message	Long Message
10001	Internal Error	Internal Error

TransactionSearch API Errors

TABLE *A.7 TransactionSearch API Errors*

Error Code	Short Message	Long Message
10001	Internal Error	Internal Error
10001	ButtonSource value truncated.	The transaction could not be loaded
10003	Transaction refused because of an invalid argument. See additional error messages for details.	Start date is a required parameter
10004	Transaction refused because of an invalid argument. See additional error messages for details.	Start date is invalid
10004	Transaction refused because of an invalid argument. See additional error messages for details.	End date is invalid
10004	Transaction refused because of an invalid argument. See additional error messages for details.	Currency is not supported
10004	Transaction refused because of an invalid argument. See additional error messages for details.	Transaction class is not supported
10004	Transaction refused because of an invalid argument. See additional error messages for details.	Receipt id is not valid
10004	Transaction refused because of an invalid argument. See additional error messages for details.	Payer email is invalid
10004	Transaction refused because of an invalid argument. See additional error messages for details.	Auction item id is not valid

TABLE A.7 TransactionSearch API Errors

Error Code	Short Message	Long Message
10004	Transaction refused because of an invalid argument. See additional error messages for details.	Receiver email is invalid
10004	Transaction refused because of an invalid argument. See additional error messages for details.	You can not search for a transaction id and a receipt id
10004	Transaction refused because of an invalid argument. See additional error messages for details.	Receiver can only be specified for payments you've received
10004	Transaction refused because of an invalid argument. See additional error messages for details.	The transaction id is not valid
10007	Permission denied	You do not have permissions to search for this transaction
10007	Permission denied	You do not have permissions to make this API call
11002	Search warning	The number of results were truncated. Please change your search parameters if you wish to see all your results.

B The Java SDK for the SOAP API

The SDK is composed of *API Services, Encrypted Website Payments (EWP) Services,* and user *profiles* that define characteristics of the users of those services. This section includes the following topics:

Installing the SDK

The PayPal SDK has been verified to work with the following standards.

TABLE **B.1** *Supported Standards*

Standard	Version
ColdFusion	7.x MX
Apache Axis	1.3
Java Runtime Environment	1.4.2 or greater
Simple Object Access Protocol (SOAP)	1.1
Web Services Description Language (WSDL)	1.1

Supported Human Languages

The PayPal SDK is available in U.S. English.

SDK Version Number

This appendix describes PayPal Java SDK version 4.3.1.

Recommended Hardware Configuration

The minimum hardware requirements for using the PayPal SDK in development and test are listed below. Production systems might require more capacity, depending on their expected load.

TABLE **B.2** **Recommended Hardware Configuration**

Component	Minimum Capacity
RAM	256 MB
CPU	Pentium 1 GHz
Disk space	50 MB

Download and Unzip the SDK

The latest version of the PayPal SDK is available at https://www.paypal.com/sdk.

1. Download the zipfile distribution.

2. Unzip the zipfile to any directory of you choose.

We refer to the directory in which you choose to extract the SDK as:

SDK_root

Post-installation Set-up

This section details steps you must take before you start using the PayPal SDK.

Adding SDK JAR Files to CLASSPATH

Before developing applications with the Java SDK, be sure to add the SDK JAR files in *SDK_root*/lib to your CLASSPATH environment variable.

SDK Directories and Optional Configurations

The Java SDK components are organized into different subdirectories, as shown in Table B.3, "PayPal SDK for Java: Directories and Contents."

TABLE B.3   PayPal SDK for Java: Directories and Contents

Directory	Descrption
cert	PayPal public certificates for Live PayPal and PayPal Sandbox
docs	SDK API documentation
lib	SDK libraries
licenses	License files
samples	Example programs that use the SDK.
src	SDK source code
tools	Third-party applications

SDK Logging

Axis Set-up for SOAP Message Logging

To enable SOAP logging with the PayPal SDK for Java, you need to configure Axis to use the `com.paypal.sdk.util.DefaultSOAPHandler` class. This is typically done through the Axis client configuration file `client-config.wsdd`, which is in the SDK `lib` directory. After the SDK is installed you should not have to do anything except to make sure the SDK resources are correctly loaded in Axis. A good example is the `run.bat` file in the `samples/SampleApp` directory of the SDK.

In addition, see the DEBUG log setting in Table B.4, "SDK Logging Levels."

NOTE: The PayPal SDK uses log4j public domain logging software. For complete information, see documentation at http://logging.apache.org/log4j/docs/.

Setting Log Levels

Set the value of the `level` element in *SDK_root*/lib/`log4j.properties`.

TABLE B.4   SDK Logging Levels

Level	Description
ALL	Same as `DEBUG`
ERROR	Log only severe errors
INFO	Date/time of API operation, operation name, elapsed time, success or failure indication

TABLE B.4   SDK Logging Levels

Level	Description
DEBUG	Full text of SOAP requests and responses and other debugging messages. Because DEBUG logging can degrade the performance of the SDK, be careful about using it for day-to-day operation. **NOTE:** Because SOAP requests and responses are asynchronous, the recording of requests and responses might appear out of sequence in the log file.

Logfile Backup

The default size of the SDK log is 10MB. You can set this size larger or smaller with the value of param name="MaxFileSize" in log4j.properties. When the log file reaches its maximum size, a backup file is created and a new logfile begins.

API Services

Integrated access to the PayPal Web Services APIs is one of the main features of the PayPal SDK. As shown in Figure B.1, "PayPal SDK API Services: High-Level View" on page 128, the PayPal SDK uses the underlying platform's SOAP toolkit to communicate with the PayPal API endpoint. Stub classes representing the requests, the responses, and their data are provided to set request parameters and read response values. These classes are auto-generated from the PayPal WSDL file.

FIGURE B.1   PayPal SDK API Services: High-Level View
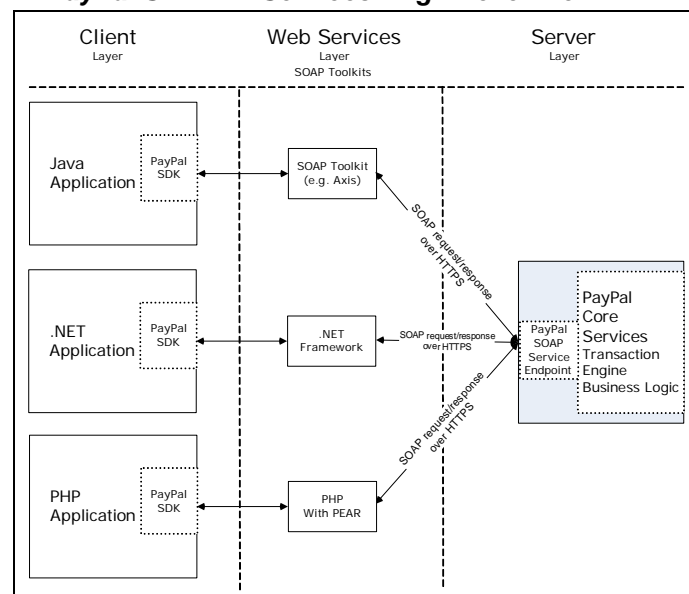
Complete SDK and API Class Documentation

Complete Javadoc

documentation for all PayPal SDK interfaces, classes, methods, structures, and data types are included with the SDK distribution. To view the documentation, open the following file with your web browser:

SDK_root/docs/index.html

Profiles

Before the SDK can be used, it must know the profile of the user accessing its services. A *profile* is a collection of information about a merchant or developer who uses the PayPal SDK. An *API profile* is associated with API Services and includes:

- A PayPal API username and password.

- If you are using API Certificates, the path to the API certificate in P12 format and the private key password to that certificate.

- If you are using API Signatures, the signature string.

- The optional name of a third-party who authorizes the caller to invoke PayPal APIs on his behalf. This third-party is called a *subject*.

- The PayPal environment for processing API calls: live, sandbox or beta-sandbox.

An EWP profile is associated with EWP Services includes:

- The path to the merchant's local copy of that public certificate

- The private key password for that public certificate

- The path to a merchant's private key file for digitally signing data

- The URL to which the button form POSTs

- The optional URL of a payment button image. The default is PayPal's standard Buy Now button.

For more information about how EWP works, see the *Website Payments Standard Integration Guide*.

Overview to Profile-related Classes

The primary interfaces and classes for SDK profiles are described below.

TABLE B.5   Summary of Java SDK Profile-related Interfaces and Classes

Interface/Class	Descriptions
APIProfile interface	This interface defines the basic information that PayPal needs to know about a user of the PayPal Web Service APIs. Developers must create an instance of APIProfile for each account that accesses the APIs. For single-merchant developers, only a single APIProfile instance is needed. PayPal provides two implementation classes suitable for the needs of most SDK developers: CertificateAPIProfile and SignatureAPIProfile. However, you are free to write a custom implementation if you need additional functionality the default classes do not offer.

TABLE B.5 ***Summary of Java SDK Profile-related Interfaces and Classes***

Interface/Class	Descriptions
EWPProfile interface	This interface defines the basic information that PayPal needs to know about a user of PayPal's Encrypted Website Payments (EWP) service. Developers must create an instance of EWPProfile for each account that generates the encrypted button code; for single-merchant users this will just be a single instance). PayPal provides a basic implementation class called DefaultEWPProfile suitable for the needs of most SDK developers. However, you are free to write a custom implementation if you need functionality the default class does not offer.
ProfileFactory class	This class creates both APIProfile and EWPProfile objects. It contains static methods that handle the instantiation and construction of profile objects.
Profiles class	This data class represents all profiles the SDK knows about. It contains two collections, one for APIProfiles and one for EWPProfiles. This class is provided to ProfileHandler to save profile data and returned from ProfileHandler to retrieve profile data.

Sample Applications

The PayPal SDK includes sample applications in the *SDK_root*/`samples` directory. Each subdirectory comes with a README file that explains how to set up the application.

TABLE B.6 ***PayPal SDK for Java: Sample Programs in SDK_root/samples***

Subdirectory	Descrption
Cert	API certificates used by the sample applications
ColdFusion	ColdFusion cfm files to call TransactionSearch, GetTransactionDetails, and RefundTransaction
JSP	JavaScript implementation for Apache Tomcat of the following PayPal APIs: • Direct Payment for final sale and for authorization • Express Checkout for final sale, authorization, and order • TransactionSearch • GetTransactionDetails • RefundTransaction • DoCapture and DoVoid • MassPay • Reauthorization

TABLE B.6   PayPal SDK for Java: Sample Programs in SDK_root/samples

Subdirectory	Descrption
SampleApp	Native Java application to call TransactionSearch, GetTransactionDetails, and Direct Payment

Sample API User with API Signature

The samples come with an API Signature for use with the SDK and the PayPal Sandbox. This API Signature belongs to the following user:

TABLE B.7   Details of the SDK Sample API Signature

API Username	`sdk-three_api1.sdk.com`
API Password	`QFZCWN5HZM8VBG7Q`
API Signature	A-IzJhZZjhg29XQ2qnhapuwxIDzyAZQ92FRP5dqBzVesOkzbdUONzmOU

Sample API User with API Certificate

The samples come with an API digital certificate for use with the SDK and the PayPal Sandbox. This certificate belongs to the following user:

TABLE B.8   Details of the SDK Sample API Certificate

Location of Certificate	*SDK_root*`\samples\Certs\sdk-seller.p12`
API Username	`sdk-seller_api1.sdk.com`
API Password	`12345678`
PKCS12 Passphrase	`password`

Configuring ColdFusion for the PayPal Java SDK

This appendix describes how to configure ColdFusion MX 7.0.1 to use the PayPal Java SDK.

Summary of Installation Steps

The ColdFusion application server must be configured to know the location of the PayPal JAR files, and your ColdFusion Markup (CFM) pages must be configured with the absolute path to the PayPal API certificate for the PayPal API user on whose behalf the calls are made.

1. Install the PayPal Java SDK as described in .

2. Copy a subset of the Java SDK JAR files to a location accessible by the ColdFusion application server. The JAR files are in *SDK_root*\lib and their exact names are as follows:

   - `bcmail-jdk14-128.jar`
   - `bcprov-jdk14-128.jar`
   - `paypal_base.jar`
   - `paypal_stubs.jar`
   - `sax2.jar`
   - `xerces.jar`
   - `xpp3-1.1.3.4d_b4_min.jar`
   - `xstream.jar-1.1.3.jar`

3. With the ColdFusion Application Server Administrator, add the absolute path of the location you determined in Step 2 to the Java and JVM `CLASSPATH` environment variable.

4. Restart the ColdFusion Application Server.

5. Copy your PKCS (P12) format PayPal API certificates to a location that you will specify in the individual CFM pages for individual API calls and users.

6. As a starting point for your own development, copy the PayPal-provided sample CFM pages to your ColdFusion application server web document root. See "Samples of CFM Pages" on page 132.

Optional SOAP Logging with ColdFusion

To configure ColdFusion to log SOAP messages, add the following lines to the `globalConfiguration` section of the *ColdFusionMX7_root_directory*\wwwroot\WEB-INF\client-confid.wsdd file.

```
<requestFlow>
    <handler type="java:com.paypal.sdk.logging.DefaultSOAPHandler" />
</requestFlow>
<responseFlow>
    <handler type="java:com.paypal.sdk.logging.DefaultSOAPHandler" />
</responseFlow>
```

Known Issue with UTF-8 Encoding in Logging. The PayPal ColdFusion sample applications use the version of log4j (1.1.3) that comes with ColdFusion MX 7. In this version of log4j, the FileAppender apparently disregards the value of the `encoding` parameter, and therefore, UTF-8 encoded characters in the SOAP messages are not logged properly.

Samples of CFM Pages

The PayPal Java SDK comes with examples of ColdFusion Markup (

CFM) pages for calling a variety of PayPal APIs. The samples are located in the following directory:

SDK_root\Samples\ColdFusion

The ColdFusion examples make the following assumptions, as specified in the set-up steps in *SDK_root*\Samples\ColdFusion\README.txt:

- ColdFusion is running on Microsoft Windows.

- The PayPal Java SDK class libraries are stored in C:\paypal\lib.

- The PayPal API user certificates are stored in C:\paypal\users.

About directory names in the samples

The samples assume certain directory locations. You can use whatever directory locations you prefer, as long as you are consistent in specifying your preferred locations in the PayPal-provided samples.

For instance, you might want to store the PKCS-format PayPal API certificates in subdirectories, one per API user. Your CFM pages for calling PayPal APIs for individual API users must be sure to use the absolute path to the correct API certificate file.

Recopying JAR Files after WSDL Updates

After updating the SDK, be sure to re-copy the PayPal JAR files to your ColdFusion installation, as detailed in "

C The ASP.NET and Classic ASP SDKs for the SOAP API

The SDK is composed of *API Services, Encrypted Website Payments (EWP) Services,* and user *profiles* that define characteristics of the users of those services. This section includes the following topics:

Installing the SDK

This section details the software and hardware supported and required by the PayPal SDK, installation, and post-installation tasks.

Supported Standards

The PayPal SDK has been verified to work with the following standards.

TABLE C.1   Supported Standards

Standard	Version
Microsoft .NET Framework	1.1, Service Pack 1
Simple Object Access Protocol (SOAP)	1.1
Web Services Description Language (WSDL)	1.1

Supported Human Languages

U.S. English

SDK Version Number

PayPal ASP.NET and Classic ASP SDKs version 4.3.1

Minimum Hardware Requirements

The following table lists the minimum hardware requirements for using the PayPal SDK in development and testing. Production systems might require more capacity, depending on their expected load.

TABLE C.2   Minimum System Hardware Requirements

Component	Minimum Capacity
RAM	256 MB
CPU	Pentium 1 GHz
Disk space	50 MB

Software Requirements

Microsoft .NET Framework 1.1, Service Pack 1

NOTE: You can download Service Pack 1 from the Microsoft web site.

Downloading and Installing the SDK

The latest version of the PayPal SDK is available at https://www.paypal.com/sdk. You can download either a self-extracting installation program or a ZIP file.

The installation is straightforward and requires no special instruction. Installing the SDK source is optional.

SDK Directories

The SDK components are organized into different subdirectories, as shown in Table C.3, "PayPal SDK Directories and Contents."

TABLE C.3   PayPal SDK Directories and Contents

Directory	Descrption
`bin`	Compiled SDK DLLs and the SDK console (`Console.exe`)
`docs`	Ndoc class documentation and SDK guide
`samples\ASPNET or ClassicASP`	Example programs that use the SDK, in subdirectories
`samples\cert`	`sdk-seller.p12` API certificate for API user `sdk-seller_api1.sdk.com`
`src`	Visual Studio project files and SDK source files. This folder is present only if you installed the source of the SDK.

Uninstalling the SDK

To uninstall the SDK, use the Microsoft Windows control panel **Add/Remove Programs**.

Installing the Samples

The SDK comes with sample applications for your study and use. These samples can be installed in Microsoft Internet Information Server. For information, see "Sample Applications" on page 144.

Configuring the SDK

This section describes how to configure the SDK to work with the sample applications and programs you develop.

Referencing the SDK DLLs for ASP.NET

Before developing applications with the SDK, be sure to add references in your ASP.NET projects to the SDK dynamic load libraries (DLLs) in *SDK_root*\bin.

Modifying the Web.config File

You can modify the PayPal ASP.NET and Classic ASP SDK behavior by making changes to the Web.config configuration file.

Initializing the Web.config File for Classic ASP

The Classic ASP SDK relies on a Web.config configuration file:

```
SDK_root\Samples\ClassicASP\Web.config
```

To enable logging for your SDK-based web applications, the SDK must know the location of its Web.config file. In your Global.asa file, modify the Session_OnStart method to call the SDK's COMAdapter.Init method:

```
Sub Session_OnStart
Set pp_caller =  CreateObject("com.paypal.sdk.COMNetInterop.COMAdapter")
pp_caller.Init Request.ServerVariables("APPL_PHYSICAL_PATH")
End Sub
```

NOTE: This step is not necessary for ASP.NET.

Optional Custom Configurations in Web.config

You can add optional custom settings to the Web.config file.

Immediately after the top-level <configuration> tag, you see a <configSections> tag. In <configSections>, add a <section name="paypal"> tag, as shown below.

```
<configuration>
```

```
<configSections>
    <section name="paypal"
type="com.paypal.sdk.core.ConfigSectionHandler, paypal_base"/>
<configSections>
```

The optional custom settings are in a `<paypal>` block later in the file:

```
<paypal>
    ... custom settings ...
</paypal>
```

Setting Maximum Retries of SOAP Requests

In the `<paypal>` block in `Web.config`, you can set the maximum number of retries to send SOAP requests.

`<maximumRetries value="`*integerNumberOfRetries*`" />`

A value of 0 for *integerNumberOfRetries* sends a SOAP request once, with no retries. The default setting is 0.

PayPal recommends setting maximumRetries no greater than 3.

SDK Logging

The PayPal SDK uses log4net public domain logging software. For information about log4net, see the log4net documentation at http://logging.apache.org/log4net/release/manual/introduction.html.

This section describes SDK logging levels, the configuration files in which you set the desired level, and SOAP request logging.

Log Levels

The SDK varies the amount of detail it records according to four logging levels.

TABLE C.4 SDK Logging Levels

Level	Description
ALL	Same as DEBUG
ERROR	Log only severe errors
INFO	Date/time of API operation, operation name, elapsed time, success or failure indication
DEBUG	Full text of SOAP requests and responses and other debugging messages. DEBUG logging can degrade the performance of the SDK. Be careful about using it for day-to-day operation.
	NOTE: Because SOAP requests and responses are asynchronous, the recording of requests and responses might appear out of sequence in the log file.

Setting SDK Log Levels

To enable logging for your SDK-based web applications, add the following lines inside the `<configuration>` block of the `Web.config` file. You can copy these lines from the *SDK_root*\samples\ASPNET\Web.config or *SDK_root*\samples\ClassicASP\Web.config file.

● You can change the value of the `file` element to write log records to a specific location.

● Set the value of the `level` element to the desired detail described in Table C.4, "SDK Logging Levels."

```
<configSections>
    <section name="log4net"
type="log4net.Config.Log4NetConfigurationSectionHandler,log4net"/>
</configSections>
<log4net>
    <appender name="PAYPALLOGFILE"
type="log4net.Appender.RollingFileAppender">
        <file value="logs/paypal.sdk.log" />
        <appendToFile value="true" />
        <encoding value="UTF-8" />
        <rollingStyle value="5" />
        <maxSizeRollBackups value="10" />
        <maximumFileSize value="10MB" />
        <staticLogFileName value="true" />
        <layout type="log4net.Layout.PatternLayout">
            <conversionPattern value="%d{dd MMM yyyy HH:mm:ss} %-5p [%C{1}]
%m%n" />
        </layout>
    </appender>
    <logger name="com.paypal.sdk"><level value="ALL" />
        <appender-ref ref="PAYPALLOGFILE" />
    </logger>
</log4net>
```

SOAP Request Logging

To enable logging of the full text of SOAP requests and responses, add the following lines inside the `<system.web>` block of your `Web.config` file. You can copy these lines from the *SDK_root*\samples\ASPNET\Web.config or *SDK_root*\samples\ClassicASP\Web.config file.

IMPORTANT: DEBUG logging can degrade the performance of the SDK. Be careful about using it for day-to-day operation.

```
<webServices>
    <soapExtensionTypes>
        <add type="com.paypal.sdk.logging.DefaultSOAPHandler, paypal_base"
priority="1" group="0" />
    </soapExtensionTypes>
</webServices>
```

Enabling Proxy Support for ASP.NET

If your application is behind a proxy server, you must enable proxy support in the Web.config file. For details on how to use the system.net element in the Web.config file, please refer to Configuring Internet Applications in the MSDN Library.

API Services

Integrated access to the PayPal Web Services APIs is one of the main features of the PayPal SDK. As shown in Figure C.1, "PayPal SDK API Services: High-Level View" on page 140, the PayPal SDK uses the underlying platform's SOAP toolkit to communicate with the PayPal API endpoint. Stub classes representing the requests, the responses, and their data are provided to set request parameters and read response values. These classes are auto-generated from the PayPal WSDL file.

FIGURE C.1 *PayPal SDK API Services: High-Level View*
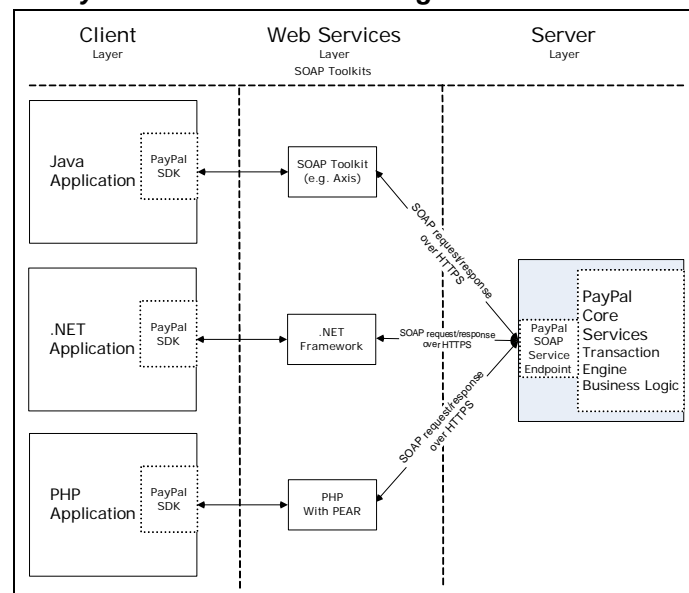
Complete SDK and API Class Documentation

Complete Microsoft .NET Ndoc documentation for all PayPal SDK interfaces, classes, methods, structures, and data types are included with the SDK distribution. To view the documentation, open the following file with your web browser:

SDK_root/docs/PayPalBaseAPI.chm

Profiles

Before the SDK can be used, it must know the profile of the user accessing its services. A *profile* is a collection of information about a merchant or developer who uses the PayPal SDK. An *API profile* is associated with API Services and includes:

- A PayPal API username and password.

- If you are using API Certificates, the path to the API certificate in P12 format and the private key password to that certificate.

- If you are using API Signatures, the signature string.

- The name of a third-party who authorizes the caller to invoke PayPal APIs on his behalf. This third-party is called a *subject*. This name is optional.

- The PayPal environment for processing API calls: `live`, `sandbox` or `beta-sandbox`.

An Encryted Wweb Payments (EWP) profile is associated with EWP services includes:

- The path to the merchant's local copy of that public certificate

- The private key password for that public certificate

- The path to a merchant's private key file for digitally signing data

- The URL to which the button form POSTs

- The optional URL of a payment button image. The default is PayPal's standard Buy Now button.

For more information about how EWP works, see the *Website Payments Standard Integration Guide*.

Overview to Profile-related Classes

The primary interfaces and classes for SDK profiles are described below.

TABLE C.5   Summary of ASP.NET and Classic ASP SDK Profile-related Interfaces and Classes

Interface/Class	Description
`IAPIProfile` interface	This interface defines the basic information that PayPal needs to know about a user of the PayPal Web Service APIs. Developers must create an instance of `IAPIProfile` for each account that accesses the APIs. For single-merchant developers, only a single `IAPIProfile` instance is needed. PayPal provides a default implementation class called `DefaultAPIProfile` suitable for the needs of most SDK developers. However, can write a custom implementation if you need additional functionality the default class does not offer.
`ProfileFactory` class	This class creates `IAPIProfile` object. It contains static methods that handle the instantiation and construction of profile objects.

TABLE C.5 *Summary of ASP.NET and Classic ASP SDK Profile-related Interfaces and Classes*

Interface/Class	Description
EWPProfile interface	This interface defines the basic information that PayPal needs to know about a user of PayPal's Encrypted Website Payments (EWP) service. Developers must create an instance of EWPProfile for each account that generates the encrypted button code for single-merchant users. This will be just be a single instance. PayPal provides a basic implementation class called DefaultEWPProfile suitable for the needs of most SDK developers. However, you can write a custom implementation if you need functionality the default class does not offer.

COM-Specific Classes

COM developers can use of the following classes:

- COMAdapter2
- COMArray
- COMUtil

These classes are in the com.paypal.sdk.COMNetInterop namespace. For examples of them in use, see the source code in SDK_Root\Samples\ClassicASP.

COMAdapter2

The COMAdapter2 class includes methods for setting and retrieving the values associated with an API profile, including the API username and password, the API signature, the third-party subject and the environment. It also includes methods for making API calls.

The older COMadapter class is for backward-compatibility.

For example, the following code snippet sets the API Profile to the credentials of the sample user supplied with the PayPal SDK, as described in "Sample Applications" on page 144." This sample user relies on an API certificate:

```
Set pp_caller = CreateObject("com.paypal.sdk.COMNetInterop.COMAdapter2")
' Set API profile
pp_caller.SetAPIUsername "sdk-seller_api1.sdk.com"
pp_caller.SetAPIPassword "12345678"
' Set the third-party subject to null
pp_caller.SetSubject ""
pp_caller.SetEnvironment "sandbox"
```

The following code snippet sets the API Profile to the credentials of a user who relies on an API signature:

```
' Set API profile
pp_caller.SetAPIUsername "seller_api1.designerfotos.com"
pp_caller.SetAPIPassword "5GM8R4SNEPLGP524"
' Set the API Signature string
pp_caller.SetAPISignature
"A0iFR8ojMS.TH8Eyc3t0CP1dTJzbA.cDbvQAR0GPJVEuvtDzkiPJNKj7"
' Set the third-party subject to null
pp_caller.SetSubject ""
pp_caller.SetEnvironment "sandbox"
```

COMArray

The COMArray proxy class functions similarly to the .NET ArrayList class. Use COMArray when you need to create a .NET array when you are setting up the request object; for example, when you provide multiple PaymentDetailsType items to a DoDirectPayment request.

For example:

```
<%
Dim pmtDetailsItemArray
Set pmtDetailsItemArray =
CreateObject("com.paypal.sdk.COMNetInterop.COMArray")
Dim pmtDetailsItem1
Set pmtDetailsItem1 =
CreateObject("com.paypal.soap.api.PaymentDetailsItemType")
pmtDetailsItemArray.Add(pmtDetailsItem1)
Dim paymentDetailsType
Set paymentDetailsType =
CreateObject("com.paypal.soap.api.PaymentDetailsType")
pmtDetailsItemArray.SetField paymentDetailsType, "PaymentDetailsItem"
%>
```

COMUtil

The COMUtil utility class facilitates populating enumerated values in request objects.

For example, suppose you want to specify US Dollars as the Currency type in a DoDirectPayment call. Without the COMUtil class, you would need to specify the integer value of the enumeration with code such as the following:

```
Set .Amount = CreateObject("com.paypal.soap.api.BasicAmountType")
.Amount.currencyID = 21
```

Instead, you can use the COMUtil class as follows:

```
Set util = CreateObject("com.paypal.sdk.COMNetInterop.COMUtil")
Set .Amount = CreateObject("com.paypal.soap.api.BasicAmountType")
.Amount.currencyID = util.GetEnumValue("CurrencyCodeType", "USD")
```

Sample Applications

The PayPal SDK includes sample applications in the *SDK_root*\samples\ASPNET or
SDK_root\samples\ClassicASP folder. The samples are divided into subfolders by
products.

TABLE C.6   Samples by Product

Subfolder in SDK_Root\samples\ASPNET or ClassicASP	Products
admin	DoCapture
	DoVoid
	GetTransactionDetails
	MassPay
	RefundTransaction
	TransactionSearch
	DoReauthorization
wppro	Express Checkout ● Final Sale ● Authorization ● Order
	Direct Payment API ● Final Sale ● Authorization

Sample API User with API Signature

The samples come with an API Signature for use with the SDK and the PayPal Sandbox. This
API Signature belongs to the following user:

TABLE C.7   Details of the SDK Sample API Signature

API Username	sdk-three_api1.sdk.com
API Password	QFZCWN5HZM8VBG7Q
API Signature	A-IzJhZZjhg29XQ2qnhapuwxIDzyAZQ92FRP5dqBzVesOkzbdUONzmOU

Sample API User and Certificate

The samples come with a PayPal Web Services API digital certificate for use with the SDK and the PayPal Sandbox. This certificate belongs to the following user:

TABLE C.8   Details of the sdk-seller.p12 SDK Sample Certificate

Location of Certificate	*SDK_root*\samples\Certs\sdk-seller.p12
API Username	sdk-seller_api1.sdk.com
API Password	12345678
PKCS12 Passphrase	password

Installing the Samples in IIS

NOTE: Be sure that you are logged in as an administrator, that IIS running, and that `WinHttpCertCfg.exe` is in your PATH.

To install the samples in Microsoft IIS:

1. Run one of the following:

SDK_root\samples\ASPNET\InstallSample.bat.

SDK_root\samples\ClassicASP\InstallSample.bat.

2. For ASP.NET To enable logging, change the permissions on the *localComputerName*\ASPNET folder to Full Control.

For Classic ASP: To enable logging, for the user account that has anonymous access, change the permissions on the logs folder to Full Control. By default, this folder is named IUSER_localComputerName.

`InstallSample.bat` does the following:

– Creates a virtual directory named `PaypalASPNETSample` or `PaypalClassicASPSample` in IIS that points to *SDK_root*\samples\ASPNET or ClassicASP.

– Loads the sample API certificate *SDK_root*\samples\Certs\sdk-seller.p12 into the Microsoft Windows system store.

– Uses the `WinHttpCertCfg.exe` command to grant unlimited access for account `Everyone` to that certificate .

Running the Samples

To run the samples, open one of the following URLs in an internet browser:

```
http://localhost/PaypalASPNETSamples
http://localhost/PaypalClassicASPSamples
```

D The PHP SDK for the SOAP API

The SDK is composed of *API Services* and user *profiles* that define characteristics of the users of those services. This section includes the following topics:

- "Configuring PHP and Apache for the SDK" on page 147
- "Supported Standards" on page 148
- "Downloading and Installing the SDK" on page 148
- "SDK Directories and Configuration Files" on page 150
- "API Services" on page 152
- "Profiles" on page 152
- "Sample Applications" on page 153

This chapter details the software and hardware supported and required by the PayPal SDK and its installation.

Configuring PHP and Apache for the SDK

The PHP SDK relies on the following required PHP packages:

- NET_Url
- NET_Socket
- HTTP_Request
- Log

Configure PHP and Apache as follows to use the PayPal SDK:

1. Disable safe mode features.

2. Do not use the disabled_functions and disabled_classes settings.

3. Install PHP as an Apache Web Server module. This instructions apply to PHP 4.3.0 or 5.1.4 on Microsoft Windows:

   – Insert the following lines into your Apache httpd.conf (which is usually located in C:\Program Files\Apache Group\Apache2\conf)

```
# For PHP 4 do something like this:
LoadModule php4_module "c:/php/php4apache2.dll"
# Don't forget to copy the php4apache2.dll file from the sapi directory!
AddType application/x-httpd-php .php
# For PHP 5 do something like this:
LoadModule php5_module "c:/php/php5apache2.dll"
```

```
AddType application/x-httpd-php .php
# configure the path to php.ini
PHPIniDir "C:/php"
```

– Go to your PHP installation directory and edit the php.ini file. Find extensions section, and uncomment (remove the semicolons at the beginning) the following two lines:

```
;extension=php_curl.dll
;extension=php_openssl.dll
```

4. Verify your PHP configuration with the following command, which displays your PHP configuration

```
php –info
```

Supported Standards

The PayPal SDK has been verified to work with the following standards.

TABLE *D.1 Supported Standards*

Standard	Version
PHP with PHP Extension and Application Repository (PEAR)	4.3.*
Simple Object Access Protocol (SOAP)	1.1
Web Services Description Language (WSDL)	1.1

Supported Human Languages

The PayPal SDK is available in U.S. English.

SDK Version Number

This guide describes PayPal PHP SDK version 4.3.1.

Downloading and Installing the SDK

The latest version of the PayPal SDK is available at https://www.paypal.com/sdk.

Be sure you have configured PHP and your web server as detailed in "Configuring PHP and Apache for the SDK" on page 147.

You can install the SDK on a local computer or on a remote computer.

Installing on Local Computer

Follow these steps to install the SDK on your local computer.

1. Download the PHP SDK in your desired distribution format: .zip, .tar, or .tar.gz.

2. Extract the distribution to your installation directory (indicated in these steps as install_dir) with one of the following commands:

```
unzip paypal_php_sdk.zip
tar vxf paypal_php_sdk.tar
tar vxfz paypal_php_sdk.tar.gz
```

3. Install the samples with the following steps:

```
cd install_dir\php-sdkphp install.php
```

4. When you are prompted for your target web directory, enter the name of your PHP web directory. The target directory must be under the document root of the PHP server.

Installing on Remote Computer from Microsoft Windows

You can install the SDK to a remote computer that runs an FTP server. You need to know the following information to complete the remote installation:

- The FTP server name
- Your FTP username and password
- The directory for the SDK itself. This must be relative to the FTP root directory.
- The directory for the SDK samples. This must be relative to the FTP root directory.

Follow these steps to install the SDK on your local computer.

1. Download a distribution format for the PHP DSK: .zip, .tar, or .tar.gz.

2. Extract the distribution to your installation directory (indicated in these steps as install_dir) with one of the following commands:

```
unzip paypal_php_sdk.zip
tar vxf paypal_php_sdk.tar
tar vxfz paypal_php_sdk.tar.gz
```

3. Install the SDK and samples with the following steps at the command prompt:

```
cd install_dir\php-sdk
remote_intall.bat
```

4. Follow the prompts until the installation finishes.

Example Remote Installation

```
C:\php-sdk>remote_install.bat
This installer will overwrite any existing files in the target directory.
To cancel, press Control-C and then Y.
```

```
 Press any key to continue . . .
Enter the FTP server to install on: yourDomainName
Username for FTP: yourFTPusername
Password for FTP: yourFTPpassword
 Please enter the path (relative to your FTP root) for the paypal-php-sdk
directory: yourRemoteInstallDirectory
 Please enter the path (relative to your FTP root) for the paypal-sdk-
samples directory: yourRemoteInstallDirectoryInThePHPDocumentRoot
```

SDK Directories and Configuration Files

The PHP SDK components are organized into different subdirectories, as shown below"

TABLE D.2   PayPal SDK for PHP: Directories and Contents

Directory	Descrption
lib	Primary SDK code. Subdirectories are described in Table D.3, "lib Subdirectories and Contents."
docs	SDK documentation: examples, sample XML, and API documentation
samples	Sample applications

TABLE D.3   lib Subdirectories and Contents

Directory in lib	Descrption
PayPal	Main SDK classes, configuration files, and ancillary files
PayPal/cert	PayPal public certificates
PayPal/conf	SDK system properties configuration file. See "Configuration File" on page 151.
PayPal/Profile	APIProfile and EWPProfile implementations and the ProfileHandler definitions, with two deafult profile handlers: ProfileHandler_File and ProfileHandler_Array.
PayPal/SOAP	PEAR::SOAP implementation. No PayPal-specific code.
PayPal/Type	Auto-generated data type files from which the PayPal type classes inherit, and two parent classes: XSDType and XSDSimpleType.
PayPal/wsdl	PayPal WSDL/XSD files and paypal-endpoints.php

Configuration File

The PayPal SDK is distributed with the configuration properties file SDK_root/lib/PayPal/conf/paypal-sdk.php.dist.

If you want to change the configuration, copy this file to:

SDK_root/lib/PayPal/conf/paypal-sdk.php

SDK Logging

The PayPal SDK uses PEAR's logging capability.

Setting Log Levels

The SDK varies the amount of detail it logs according to three logging levels.

Set the value of the log_level parameter in the SDK_root/lib/PayPal/conf/paypal-sdk.php file.

TABLE D.4   SDK Logging Levels

Level	Description
PEAR_LOG_ERR	Log only severe errors
PEAR_LOG_INFO	Default log level. Date/time of API operation, operation name, elapsed time, success or failure indication.
PEAR_LOG_DEBUG	Full text of SOAP requests and responses and other debugging messages. Because SOAP requests and responses are asynchronous, the recording of requests and responses might appear out of sequence in the log file. Because DEBUG logging can degrade the performance of the SDK, be careful about using it for day-to-day operation.

TABLE D.5   SDK Logging Levels

Level	Description
ALL	Same as DEBUG
ERROR	Log only severe errors
INFO	Date/time of API operation, operation name, elapsed time, success or failure indication
DEBUG	Full text of SOAP requests and responses and other debugging messages. Because DEBUG logging can degrade the performance of the SDK, be careful about using it for day-to-day operation. **NOTE:** Because SOAP requests and responses are asynchronous, the recording of requests and responses might appear out of sequence in the log file.

Logfile Location

In the log_dir parameter, set the name of the directory where the SDK should write its logfile.

API Services

Integrated access to the PayPal Web Services APIs is one of the main features of the PayPal SDK. As shown in Figure D.1, "PayPal SDK API Services: High-Level View" on page 152, the PayPal SDK uses the underlying platform's SOAP toolkit to communicate with the PayPal API endpoint. Stub classes representing the requests, the responses, and their data are provided to set request parameters and read response values. These classes are auto-generated from the PayPal WSDL file.

FIGURE D.1 PayPal SDK API Services: High-Level View
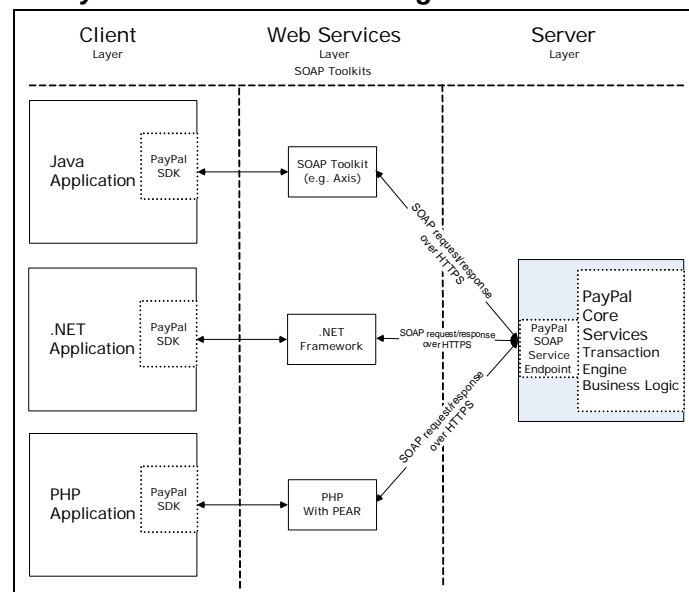
Profiles

Before the SDK can be used, it must know the profile of the user accessing its services. A *profile* is a collection of information about a merchant or developer who uses the PayPal SDK. An *API profile* is associated with API Services and includes:

- A PayPal API username and password.

- If you are using API Certificates, The path to the API certificate in PEM format as downloaded from PayPal

- If you are using API Signatures, the signature string.

- The optional name of a third-party who authorizes the caller to invoke PayPal APIs on his behalf. This third-party is called a *subject*.

- The PayPal environment for processing API calls: `live`, `sandbox` or `beta-sandbox`.

An EWP profile is associated with EWP Services includes:

- The path to the merchant's local copy of that public certificate

- The private key password for that public certificate

- The path to a merchant's private key file for digitally signing data

- The URL to which the button form POSTs

- The optional URL of a payment button image. The default is PayPal's standard Buy Now button.

For more information about how EWP works, see the Website Payments Standard Integration Guide.

Overview to Profile-related Classes

The primary interfaces and classes for SDK profiles are described below.

TABLE D.6   *Summary of PHP SDK Profile-related Classes*

Interface/Class	Descriptions
APIProfile	This class defines the basic information that PayPal needs to know about a user of the PayPal Web Service APIs. Developers must create an instance of APIProfile for each account that accesses the APIs. For single-merchant developers, only a single APIProfile instance is needed. PayPal provides a default implementation suitable for the needs of most SDK developers. However, you are free to write a custom implementation if you need additional functionality the default class does not offer.
ProfileHandler	The ProfileHandler class manages the storage and retrieval of APIProfile and EWPProfile objects. PayPal provides two basic implementations: ProfileHandler_Array and ProfileHandler_File to serialize and deserialize the data in profiles to and from a file on the disk. For security, the ProfileHandler_File implementation does not store an API user's API password.

Sample Applications

The PayPal SDK includes sample applications in the *SDK_root*/samples. The sample applications are divided into two groups:

1. Website Payments Pro: Using the Direct Payment API for credit card charges and the Express Checkout APIs

2. Administrative Samples: RefundTransaction, TransactionSearch, DoCapture, DoVoid, and DoReauthorization

Sample API User with API Signature

The samples come with an API Signature for use with the SDK and the PayPal Sandbox. This API Signature belongs to the following user:

TABLE D.7   Details of the SDK Sample API Signature

API Username	`sdk-three_api1.sdk.com`
API Password	QFZCWN5HZM8VBG7Q
API Signature	A-IzJhZZjhg29XQ2qnhapuwxIDzyAZQ92FRP5dqBzVesOkzbdUONzmOU

Sample API User and Certificate

The samples come with a PayPal Web Services API digital certificate for use with the SDK and the PayPal Sandbox. This certificate belongs to the following user:

TABLE D.8   Details of the sdk-seller.p12 SDK Sample Certificate

Location of Certificate	*SDK_root*\samples\Certs\sdk-seller.p12
API Username	`sdk-seller_api1.sdk.com`
API Password	`12345678`
PKCS12 Passphrase	`password`

Running the Samples

Run the samples by opening the following URL:

`http://yourWebServer:port/webDir/paypal_php_samples/`

webDir will vary depending on the target PHP Web directory you specified to install.php when you installed the SDK.

B State and Province Abbreviations

TABLE B.1 Abbreviations for Canadian Provinces and U.S. States

Canadian Province or U.S. State	Abbreviation
Alberta	AB
British Columbia	BC
Manitoba	MB
New Brunswick	NB
Newfoundland and Labrador	NF
Northwest Territories	NT
Nova Scotia	NS
Nunavut	NU
Ontario	ON
Prince Edward Island	PE
Quebec	QC
Saskatchewan	SK
Yukon	YK
Alabama	AL
Alaska	AK
American Samoa	AS
Arizona	AZ
Arkansas	AR
California	CA
Colorado	CO
Connecticut	CT
Delaware	DE
District of Columbia	DC
Federated States of Micronesia	FM

TABLE B.1 Abbreviations for Canadian Provinces and U.S. States

Canadian Province or U.S. State	Abbreviation
Florida	FL
Georgia	GA
Guam	GU
Hawaii	HI
Idaho	ID
Illinois	IL
Indiana	IN
Iowa	IA
Kansas	KS
Kentucky	KY
Louisiana	LA
Maine	ME
Marshall Islands	MH
Maryland	MD
Massachusetts	MA
Michigan	MI
Minnesota	MN
Mississippi	MS
Missouri	MO
Montana	MT
Nebraska	NE
Nevada	NV
New Hampshire	NH
New Jersey	NJ
New Mexico	NM
New York	NY
North Carolina	NC
North Dakota	ND

TABLE B.1 Abbreviations for Canadian Provinces and U.S. States

Canadian Province or U.S. State	Abbreviation
Northern Mariana Islands	MP
Ohio	OH
Oklahoma	OK
Oregon	OR
Palau	PW
Pennsylvania	PA
Puerto Rico	PR
Rhode Island	RI
South Carolina	SC
South Dakota	SD
Tennessee	TN
Texas	TX
Utah	UT
Vermont	VT
Virgin Islands	VI
Virginia	VA
Washington	WA
West Virginia	WV
Wisconsin	WI
Wyoming	WY
Armed Forces Americas	AA
Armed Forces	AE
Armed Forces Pacific	AP

F Country Codes

Table 1: Country Codes

Country	Code
AFGHANISTAN	AF
ÅLAND ISLANDS	AX
ALBANIA	AL
ALGERIA	DZ
AMERICAN SAMOA	AS
ANDORRA	AD
ANGOLA	AO
ANGUILLA	AI
ANTARCTICA	AQ
ANTIGUA AND BAR-BUDA	AG
ARGENTINA	AR
ARMENIA	AM
ARUBA	AW
AUSTRALIA	AU
AUSTRIA	AT
AZERBAIJAN	AZ
BAHAMAS	BS
BAHRAIN	BH
BANGLADESH	BD
BARBADOS	BB

Country	Code
BELARUS	BY
BELGIUM	BE
BELIZE	BZ
BENIN	BJ
BERMUDA	BM
BHUTAN	BT
BOLIVIA	BO
BOSNIA AND HERZE-GOVINA	BA
BOTSWANA	BW
BOUVET ISLAND	BV
BRAZIL	BR
BRITISH INDIAN OCEAN TERRITORY	IO
BRUNEI DARUSSALAM	BN
BULGARIA	BG
BURKINA FASO	BF
BURUNDI	BI
CAMBODIA	KH
CAMEROON	CM
CANADA	CA
CAPE VERDE	CV
CAYMAN ISLANDS	KY

Country	Code
CENTRAL AFRICAN REPUBLIC	CF
CHAD	TD
CHILE	CL
CHINA	CN
CHRISTMAS ISLAND	CX
COCOS (KEELING) ISLANDS	CC
COLOMBIA	CO
COMOROS	KM
CONGO	CG
CONGO, THE DEMO-CRATIC REPUBLIC OF THE	CD
COOK ISLANDS	CK
COSTA RICA	CR
COTE D'IVOIRE	CI
CROATIA	HR
CUBA	CU
CYPRUS	CY
CZECH REPUBLIC	CZ
DENMARK	DK
DJIBOUTI	DJ
DOMINICA	DM
DOMINICAN REPUBLIC	DO
ECUADOR	EC
EGYPT	EG
EL SALVADOR	SV

Country	Code
EQUATORIAL GUINEA	GQ
ERITREA	ER
ESTONIA	EE
ETHIOPIA	ET
FALKLAND ISLANDS (MALVINAS)	FK
FAROE ISLANDS	FO
FIJI	FJ
FINLAND	FI
FRANCE	FR
FRENCH GUIANA	GF
FRENCH POLYNESIA	PF
FRENCH SOUTHERN TERRITORIES	TF
GABON	GA
GAMBIA	GM
GEORGIA	GE
GERMANY	DE
GHANA	GH
GIBRALTAR	GI
GREECE	GR
GREENLAND	GL
GRENADA	GD
GUADELOUPE	GP
GUAM	GU
GUATEMALA	GT
GUERNSEY	GG

Country	Code	Country	Code
GUINEA	GN	KOREA, REPUBLIC OF	KR
GUINEA-BISSAU	GW	KUWAIT	KW
GUYANA	GY	KYRGYZSTAN	KG
HAITI	HT	LAO PEOPLE'S DEMO-CRATIC REPUBLIC	LA
HEARD ISLAND AND MCDONALD ISLANDS	HM	LATVIA	LV
HOLY SEE (VATICAN CITY STATE)	VA	LEBANON	LB
		LESOTHO	LS
HONDURAS	HN	LIBERIA	LR
HONG KONG	HK	LIBYAN ARAB JAMA-HIRIYA	LY
HUNGARY	HU		
ICELAND	IS	LIECHTENSTEIN	LI
INDIA	IN	LITHUANIA	LT
INDONESIA	ID	LUXEMBOURG	LU
IRAN, ISLAMIC REPUB-LIC OF	IR	MACAO	MO
IRAQ	IQ	MACEDONIA, THE FORMER YUGOSLAV REPUBLIC OF	MK
IRELAND	IE		
ISLE OF MAN	IM	MADAGASCAR	MG
ISRAEL	IL	MALAWI	MW
ITALY	IT	MALAYSIA	MY
JAMAICA	JM	MALDIVES	MV
JAPAN	JP	MALI	ML
JERSEY	JE	MALTA	MT
JORDAN	JO	MARSHALL ISLANDS	MH
KAZAKHSTAN	KZ	MARTINIQUE	MQ
KENYA	KE	MAURITANIA	MR
KIRIBATI	KI	MAURITIUS	MU
KOREA, DEMOCRATIC PEOPLE'S REPUBLIC OF	KP	MAYOTTE	YT
		MEXICO	MX

Country	Code	Country	Code
MICRONESIA, FEDER-ATED STATES OF	FM	PALAU	PW
MOLDOVA, REPUBLIC OF	MD	PALESTINIAN TERRI-TORY, OCCUPIED	PS
MONACO	MC	PANAMA	PA
MONGOLIA	MN	PAPUA NEW GUINEA	PG
MONTSERRAT	MS	PARAGUAY	PY
MOROCCO	MA	PERU	PE
MOZAMBIQUE	MZ	PHILIPPINES	PH
MYANMAR	MM	PITCAIRN	PN
NAMIBIA	NA	POLAND	PL
NAURU	NR	PORTUGAL	PT
NEPAL	NP	PUERTO RICO	PR
NETHERLANDS	NL	QATAR	QA
NETHERLANDS ANTI-LLES	AN	REUNION	RE
		ROMANIA	RO
NEW CALEDONIA	NC	RUSSIAN FEDERATION	RU
NEW ZEALAND	NZ	RWANDA	RW
NICARAGUA	NI	SAINT HELENA	SH
NIGER	NE	SAINT KITTS AND NEVIS	KN
NIGERIA	NG	SAINT LUCIA	LC
NIUE	NU	SAINT PIERRE AND MIQUELON	PM
NORFOLK ISLAND	NF		
NORTHERN MARIANA ISLANDS	MP	SAINT VINCENT AND THE GRENADINES	VC
NORWAY	NO	SAMOA	WS
OMAN	OM	SAN MARINO	SM
PAKISTAN	PK		

Country	Code
SAO TOME AND PRINCIPE	ST
SAUDI ARABIA	SA
SENEGAL	SN
SERBIA AND MONTENEGRO	CS
SEYCHELLES	SC
SIERRA LEONE	SL
SINGAPORE	SG
SLOVAKIA	SK
SLOVENIA	SI
SOLOMON ISLANDS	SB
SOMALIA	SO
SOUTH AFRICA	ZA
SOUTH GEORGIA AND THE SOUTH SANDWICH ISLANDS	GS
SPAIN	ES
SRI LANKA	LK
SUDAN	SD
SURINAME	SR
SVALBARD AND JAN MAYEN	SJ
SWAZILAND	SZ
SWEDEN	SE
SWITZERLAND	CH
SYRIAN ARAB REPUBLIC	SY
TAIWAN, PROVINCE OF CHINA	TW
TAJIKISTAN	TJ

Country	Code
TANZANIA, UNITED REPUBLIC OF	TZ
THAILAND	TH
TIMOR-LESTE	TL
TOGO	TG
TOKELAU	TK
TONGA	TO
TRINIDAD AND TOBAGO	TT
TUNISIA	TN
TURKEY	TR
TURKMENISTAN	TM
TURKS AND CAICOS ISLANDS	TC
TUVALU	TV
UGANDA	UG
UKRAINE	UA
UNITED ARAB EMIRATES	AE
UNITED KINGDOM	GB
UNITED STATES	US
UNITED STATES MINOR OUTLYING ISLANDS	UM
URUGUAY	UY
UZBEKISTAN	UZ
VANUATU	VU
VENEZUELA	VE
VIET NAM	VN
VIRGIN ISLANDS, BRITISH	VG

Country	Code
VIRGIN ISLANDS, U.S.	VI
WALLIS AND FUTUNA	WF
WESTERN SAHARA	EH
YEMEN	YE
ZAMBIA	ZM
ZIMBABWE	ZW

Index