



Adaptive Payments

Developer Guide

Last updated: January 5, 2011

Adaptive Payments Developer Guide

Document Number: 10097.en_US-20110105

© 2010-2011 PayPal, Inc. All rights reserved. PayPal is a registered trademark of PayPal, Inc. The PayPal logo is a trademark of PayPal, Inc. Other trademarks and brands are the property of their respective owners.

The information in this document belongs to PayPal, Inc. It may not be used, reproduced or disclosed without the written approval of PayPal, Inc. Copyright © PayPal. All rights reserved. PayPal S.à r.l. et Cie, S.C.A., Société en Commandite par Actions. Registered office: 22-24 Boulevard Royal, L-2449, Luxembourg, R.C.S. Luxembourg B 118 349

Consumer advisory: The PayPal™ payment service is regarded as a stored value facility under Singapore law. As such, it does not require the approval of the Monetary Authority of Singapore. You are advised to read the terms and conditions carefully.

Notice of non-liability:

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

Contents

Chapter	Preface	13
1.6.0 Features	13	
New API Operations for Version 1.6.0	13	
Changes to PayRequest Fields for Version 1.6.0	13	
Changes to PayResponse Fields for Version 1.6.0	13	
Changes to ExecutePaymentRequest Fields for Version 1.6.0	14	
Changes to GetPaymentOptionsResponse Fields for Version 1.6.0	14	
Changes to SetPaymentOptionsRequest Fields for Version 1.6.0	14	
Changes to PreapprovalRequest Fields for Version 1.6.0	14	
Changes to Address Structure for Version 1.6.0	15	
Changes to DisplayOptions Structure for Version 1.6.0	15	
New CurrencyConversion Structure for Version 1.6.0	15	
New InvoiceData Structure for Version 1.6.0	15	
New InvoiceItem Structure for Version 1.6.0	16	
New SenderOptions Structure for Version 1.6.0	16	
New SenderIdentifier Structure for Version 1.6.0	16	
New ReceiverOptions Structure for Version 1.6.0	17	
New ReceiverIdentifier Structure for Version 1.6.0	17	
Additional Error Messages for Version 1.6.0	17	
Revision History	20	
Chapter 1	Introducing Adaptive Payments	21
Adaptive Payments API Operations	21	
API Operations Related to Payments	21	
API Operations Related to Preapprovals	21	
Other API Operations	22	
Adaptive Payments Actors and Objects	22	
Simple, Parallel, and Chained Payments	24	
Simple Payments	25	
Parallel Payments	25	
Chained Payments	26	
Payment Approval	27	
Adaptive Payments Flows	28	
Explicit Approval Payments Flow	28	

Preapproved Payments Flow	30
Implicit Approval Payments Flow	31
Adaptive Payments Service Permissions	32
Embedded Payments	33
Embedded Payment Flow Presentations	33
Kinds of Embedded Payments.	35
Embedded Payments Implementation Basics	36
Embedded Payment Experience	36
Setting Up Web Pages to Invoke the Embedded Payment Flow	47
Displaying and Collecting Shipping Addresses	49
Guest Payments	49
Fee Payment Configuration	50
Sender Pays the Fee	51
Receiver Pays the Fee in a Parallel Payment	52
Each Receiver Pays the Fee in a Chained Payment	52
Primary Receiver Pays the Fee in a Chained Payment	53
Secondary Receivers Pay the Fee in a Chained Payment	54
Chapter 2 Getting Started	57
Adaptive Payments API Operations	57
API Operations Related to Payments	57
API Operations Related to Preapprovals	57
Other API Operations	58
Adaptive Payments Endpoints	58
HTTP Headers	58
Authentication	59
Specifying JSON, NVP, or XML Data Formats.	59
SOAP Messages.	60
Specifying Application and Device Information	60
Making a Simple Payment (JSON).	61
Pay Request for Simple Payment	61
Pay Response for Simple Payment	61
Redirecting Sender for Payment Approval.	62
Making a Parallel Payment (NVP)	62
Pay Request for Parallel Payment.	62
Pay Response for Parallel Payment.	63
Redirecting Sender for Payment Approval.	63
Making a Chained Payment (XML)	63
Pay Request for Chained Payment	64

Pay Response for Chained Payment	64
Redirecting Sender for Payment Approval.	64
Chapter 3 Pay API Operation	65
Pay Summary	65
Common Fields for All Payments	65
Parallel Payments	66
Chained Payments.	66
Implicit Payments	67
Preapproved Payments	67
Payments for Digital Goods	67
Payment Notifications	67
Additional Notes About the Pay API Operation	68
PayRequest Message	69
PayRequest Fields.	70
ClientDetails Fields	73
FundingConstraint Fields	74
FundingTypeList Fields	74
FundingTypeInfo Fields	75
ReceiverList Fields.	75
Receiver Fields	75
PhoneNumberType Fields	76
SenderIdentifier Fields.	77
RequestEnvelope Fields	77
PayResponse Message	77
PayResponse Fields.	81
PayErrorList Fields.	82
FundingPlan Fields	82
CurrencyType Fields.	82
FundingSource Fields	83
CurrencyConversion Fields	84
FundingPlanCharge Fields	84
PayError Fields	84
ErrorData Fields	85
Receiver Fields	85
PhoneNumberType Response Fields	86
ResponseEnvelope Fields	87
PPFault Message	87
FaultMessage Fields.	88
ErrorData Fields	88

ResponseEnvelope Fields	89
Pay Examples Using NVP and CURL	89
Pay Errors	95
Chapter 4 PaymentDetails API Operation	99
PaymentDetailsRequest Message	99
PaymentDetailsRequest Fields	99
RequestEnvelope Fields	100
PaymentDetailsResponse Message	100
PaymentDetailsResponse Fields	102
FundingTypeList Fields	105
FundingTypeInfo Fields	105
PaymentInfoList Fields	106
PaymentInfo Fields	106
Receiver Fields	107
PhoneNumberType Response Fields	108
ResponseEnvelope Fields	109
PPFault Message	109
FaultMessage Fields	110
ErrorData Fields	110
ResponseEnvelope Fields	111
PaymentDetails Examples Using NVP and CURL	111
Payment Details Errors	112
Chapter 5 ExecutePayment API Operation	115
ExecutePaymentRequest Message	115
ExecutePaymentRequest Fields	115
RequestEnvelope Fields	116
ExecutePaymentResponse Message	116
ExecutePaymentResponse Fields	118
PayErrorList Fields	118
PayError Fields	118
ErrorData Fields	119
Receiver Fields	119
ResponseEnvelope Fields	120
PPFault Message	120
FaultMessage Fields	121
ErrorData Fields	122

ResponseEnvelope Fields	122
ExecutePayment Errors	123
Chapter 6 GetPaymentOptions API Operation	127
GetPaymentOptionsRequest Message	127
GetPaymentOptionsRequest Fields	127
RequestEnvelope Fields	127
GetPaymentOptionsResponse Message	128
GetPaymentOptionsResponse Fields	129
DisplayOptions Fields	130
InitiatingEntity Fields	130
Institution Customer Fields	131
SenderOptions Fields	131
ReceiverOptions Fields	132
InvoiceData Fields	132
Invoiceltem Fields	132
ReceiverIdentifier Fields	133
PhoneNumberType Response Fields	133
ResponseEnvelope Fields	133
PPFault Message	134
FaultMessage Fields	134
ErrorData Fields	135
ResponseEnvelope Fields	135
GetPaymentOptions Errors	136
Chapter 7 SetPaymentOptions API Operation	137
SetPaymentsOptionsRequest Message	137
SetPaymentOptionsRequest Fields	139
DisplayOptions Fields	139
InitiatingEntity Fields	140
Institution Customer Fields	140
SenderOptions Fields	141
ReceiverOptions Fields	141
InvoiceData Fields	141
Invoiceltem Fields	142
ReceiverIdentifier Fields	142
PhoneNumberType Fields	142
RequestEnvelope Fields	143
SetPaymentOptionsResponse Message	143

ResponseEnvelope Fields	143
PPFault Message	144
FaultMessage Fields	144
ErrorData Fields	145
ResponseEnvelope Fields	145
SetPaymentOptions Errors	146
Chapter 8 Preapproval API Operation	149
Preapproval Overview	149
Preapproval Notifications	149
Additional Notes About the PreApproval API Operation.	150
PreapprovalRequest Message.	150
PreapprovalRequest Fields	151
ClientDetails Fields	154
RequestEnvelope Fields	155
PreapprovalResponse Message	155
PreapprovalResponse Fields	156
ResponseEnvelope Fields	156
PPFault Message	156
FaultMessage Fields.	157
ErrorData Fields	158
ResponseEnvelope Fields	158
Preapproval Examples Using NVP and CURL.	159
Preapproval Errors	160
Chapter 9 PreapprovalDetails API Operation.	161
PreapprovalDetailsRequest Message	161
PreapprovalDetailsRequest Fields.	161
RequestEnvelope Fields.	162
PreapprovalDetailsResponse Message	162
PreapprovalDetailsResponse Fields.	164
AddressList Fields	167
Address Fields	167
BaseAddress Fields	167
ResponseEnvelope Fields.	168
PPFault Message	168
FaultMessage Fields.	169
ErrorData Fields	169

ResponseEnvelope Fields	170
PreapprovalDetails Examples Using NVP and CURL	170
Preapproval Details Errors	171
Chapter 10 CancelPreapproval API Operation	173
CancelPreapprovalRequest Message	173
CancelPreapprovalRequest Fields	173
RequestEnvelope Fields	173
CancelPreapprovalResponse Message	174
CancelPreapprovalResponse Fields	174
ResponseEnvelope Fields	174
PPFault Message	175
FaultMessage Fields	176
ErrorData Fields	176
ResponseEnvelope Fields	177
CancelPreapproval Errors	177
Chapter 11 Refund API Operation	179
Refund API Overview	179
Refund Notifications	180
Additional Notes About the Refund API Operation	180
RefundRequest Message	181
RefundRequest Fields	182
ReceiverList Fields	184
Receiver Fields	184
PhoneNumberType Fields	185
RequestEnvelope Fields	185
Refund Response Message	185
RefundResponse Fields	188
RefundInfoList Fields	189
RefundInfo Fields	189
ErrorData Fields	191
Receiver Fields	191
PhoneNumberType Response Fields	192
ResponseEnvelope Fields	192
PPFault Message	193
FaultMessage Fields	193
ErrorData Fields	194

ResponseEnvelope Fields	194
Refund Examples Using NVP and CURL	195
Refund Errors	196
Chapter 12 ConvertCurrency API Operation	199
ConvertCurrencyRequest Message	199
ConvertCurrencyRequest Fields	199
CurrencyType Fields	200
CurrencyList Fields	202
CurrencyCodeList Fields	203
RequestEnvelope Fields	203
ConvertCurrencyResponse Message	204
ConversionCurrencyResponse Fields	205
CurrencyConversionTable Fields	205
CurrencyConversionList Fields	205
CurrencyType Fields	205
CurrencyList Fields	206
ResponseEnvelope Fields	207
PPFault Message	207
FaultMessage Fields	208
ErrorData Fields	208
ResponseEnvelope Fields	209
ConvertCurrency Errors	209
Chapter 13 GetFundingPlans API Operation	211
GetFundingPlans Overview	211
Additional Notes About the Pay API Operation	211
GetFundingPlansRequest Message	211
GetFundingPlansRequest Fields	212
RequestEnvelope Fields	212
GetFundingPlansResponse Message	212
GetFundingPlansResponse Fields	214
FundingPlan Fields	214
CurrencyType Fields	215
FundingSource Fields	216
CurrencyConversion Fields	216
ResponseEnvelope Fields	216
PPFault Message	217

FaultMessage Fields218
ErrorData Fields218
ResponseEnvelope Fields219
GetFundingPlan Errors219
Chapter 14 GetShippingAddresses API Operation	221
GetShippingAddressesRequest Message221
GetShippingAddressesRequest Fields221
RequestEnvelope Fields222
GetShippingAddressesResponse Message222
GetShippingAddressesResponse Fields223
Address Fields223
BaseAddress Fields223
ResponseEnvelope Fields224
PPFault Message224
FaultMessage Fields225
ErrorData Fields225
ResponseEnvelope Fields226
GetShippingAddresses Errors226
Chapter 15 Adaptive Payment Commands	229
_ap-payment Command229
_ap-preapproval Command230
Embedded Payment Flow Command230
JavaScript Functions for Embedded Payments231
Chapter 16 Instant Payment Notifications	233
Pay Message Variables233
Preapproval Message Variables236
Chapter 17 Older Versions of the Adaptive Payments API	239
1.5.0 Features239
Ability to Delay a Chained Payment Feature in Version 1.6.0239
PaymentDetails Pending Reason Feature in Version 1.6.0239
1.4.0 Features240
New API Operations for Version 1.4.0240
Changes to the Pay API Operation241

Changes to the PaymentDetails API Operation	241
Changes to the PreapprovalDetails API Operation	243
Changes to the PreapprovalDetails API Operation	243
Changes to the Refund API Operation	244
1.3.0 Features	244
Overview of Changes for Version 1.3.0	244
HTTP Header Changes	245
Changes to the Pay API Operation for Version 1.3.0	246
Changes to the PaymentDetails API Operation for Version 1.3.0	248
1.2.0 Features	249
Changes to the Pay API Operation for Version 1.2.0	250
Changes to the Payment Details API Operation for Version 1.2.0	250
Changes to Preapproval API Operation for Version 1.2.0	251
Changes to Preapproval Details API Operation for Version 1.2.0	251
1.1.0 Features	251
Changes to the Pay API Operation for Version 1.1.0	252
Changes to PaymentDetails API Operation for Version 1.1.0	252
Changes to Preapproval API Operation for Version 1.1.0	254
Changes to PreapprovalDetails API Operation for Version 1.1.0	254
Changes to Refund API Operation for Version 1.1.0	254

Preface

Adaptive Payments is intended for developers implementing payment solutions.

1.6.0 Features

Version 1.6.0 of the Adaptive Payments API introduces new API operations, as well as changes to several Adaptive Payments API operations and additional error messages.

NOTE: Changes to API operations are backward-compatible.

New API Operations for Version 1.6.0

API Operation	Description
GetFundingPlans	Determines the funding sources that are available for a specified payment
GetShippingAddresses	Obtains the selected shipping address

Changes to PayRequest Fields for Version 1.6.0

Field	Description
sender	New field: ap:SenderIdentifier <i>(Optional)</i> Sender's identifying information.

Changes to PayResponse Fields for Version 1.6.0

Field	Description
defaultFundingPlan	New field: ap:FundingPlan Default funding plan.

Changes to ExecutePaymentRequest Fields for Version 1.6.0

Field	Description
fundingPlanId	New field: xs:string <i>(Optional)</i> The ID of the funding plan from which to make this payment.

Changes to GetPaymentOptionsResponse Fields for Version 1.6.0

Field	Description
shippingAddressId	New field: xs:string Sender's shipping address ID.

Changes to SetPaymentOptionsRequest Fields for Version 1.6.0

Field	Description
shippingAddressId	New field: xs:string <i>(Optional)</i> Sender's shipping address ID.

Changes to PreapprovalRequest Fields for Version 1.6.0

Field	Description
endingDate	xs:dateTime <i>(Optional)</i> Last date for which the preapproval is valid. It cannot be later than one year from the starting date. You must specify a value unless you have specific permission from PayPal.
maxNumberOfPaymentsPerPeriod	xs:int <i>(Optional)</i> The preapproved maximum number of all payments per period. You must specify a value unless you have specific permission from PayPal.

Changes to Address Structure for Version 1.6.0

Field	Description
addressId	<p>New field: <code>xs:string</code></p> <p>The ID associated with the address.</p>

Changes to DisplayOptions Structure for Version 1.6.0

Field	Description
headerImageUrl	<p>New field: <code>xs:string</code></p> <p>(Optional) The URL of the image that displays in the header of a payment page. If set, it overrides the header image URL specified in your account's Profile. The image dimensions are 750 pixels high x 90 pixels wide.</p>
businessName	<p>New field: <code>xs:string</code></p> <p>(Optional) The business name to display.</p>

New CurrencyConversion Structure for Version 1.6.0

Field	Description
from	<code>ap:CurrencyType</code> The currency to be converted.
to	<code>ap:CurrencyType</code> The currency resulting from the conversion.
exchangeRate	<code>xs:decimal</code> The exchange rate for the <code>from</code> currency to the <code>to</code> currency.

New InvoiceData Structure for Version 1.6.0

Field	Description
item	<code>ap:InvoiceItem</code> (Optional) Any number of items associated with the payment.
totalTax	<code>xs:decimal</code> (Optional) Total tax associated with the payment.

Preface

1.6.0 Features

Field	Description
totalShipping	xs:decimal <i>(Optional)</i> Total shipping charge associated with the payment.

New InvoiceItem Structure for Version 1.6.0

Field	Description
name	xs:string <i>(Optional)</i> Name of item.
identifier	xs:string <i>(Optional)</i> External reference to item or item ID.
price	xs:decimal <i>(Optional)</i> Price of item.

New SenderOptions Structure for Version 1.6.0

Field	Description
sharePhoneNumber	xs:boolean <i>(Optional)</i> If true, share the phone number with receivers; default is false.
shareAddress	xs:boolean <i>(Optional)</i> If true, share the address with receivers; default is false.
requireShippingAddressSelection	xs:boolean <i>(Optional)</i> If true, require the sender to select a shipping address during the payment flow; default is false.

New SenderIdentifier Structure for Version 1.6.0

Field	Description
email	xs:string <i>(Optional)</i> Sender's email address. Maximum length: 127 characters
phone	common:PhoneNumberType <i>(Optional)</i> Sender's phone number.
useCredentials	xs:boolean <i>(Optional)</i> If true, use credentials to identify the sender; default is false.

New ReceiverOptions Structure for Version 1.6.0

Field	Description
description	xs:string <i>(Optional)</i> A description you want to associate with the payment.
customId	xs:string <i>(Optional)</i> An external reference or identifier you want to associate with the payment.
invoiceData	ap:InvoiceData <i>(Optional)</i> Item information for the payment, which could appear on an invoice.
receiver	ap:ReceiverIdentifier <i>(Optional)</i> Receiver ID associated with these options.

New ReceiverIdentifier Structure for Version 1.6.0

Field	Description
email	xs:string <i>(Optional)</i> Receiver's email address. Maximum length: 127 characters
phone	common:PhoneNumberType <i>(Optional)</i> Receiver's phone number.

Additional Error Messages for Version 1.6.0

Code	Message
520003	Authentication failed. API credentials are incorrect.
520003	<ul style="list-style-type: none"> • The caller's credential doesn't have permission for the fee payer <value> • The caller's credential doesn't have permission for the payment type <value> • Authentication failed. API credentials are incorrect.
540031	You do not have permission to execute this payment Operation is not permitted because the credentials do not match those of the initial operation
540031	You do not have permission to get these preapproval details
540031	You do not have permission to get these payment options Operation is not permitted because the credentials do not match those of the initial operation

Code	Message
540031	You do not have permission to set these payment options Operation is not permitted because the credentials do not match those of the initial operation
540031	Operation is not permitted because the credentials do not match those of the initial operation
540031	You do not have permission to get these payment details
550001	<ul style="list-style-type: none"> • You are not allowed to confirm the preapproval for this key • User is not allowed to perform this action.
550001	You do not have permission to execute this payment implicitly
550001	<ul style="list-style-type: none"> • User is not allowed to perform this action. • You are not allowed to get the funding plans for this key • You do not have permission to get these funding plan details
550001	<ul style="list-style-type: none"> • User is not allowed to perform this action. • You are not allowed to get the addresses for this key
580001	<ul style="list-style-type: none"> • Invalid request: both email and phone cannot be set <value> <value> <value> • Invalid request: neither email nor phone is set <value> • To use currency <value>, currency must be held by sender and all receivers • Invalid request: More than one field cannot be used to specify a sender • Invalid request: <value>
580001	Invalid request: <value>
580022	<ul style="list-style-type: none"> • Invalid request parameter: <value> • The clientDetails parameter is missing • The PIN is invalid
580022	<ul style="list-style-type: none"> • Invalid request parameter: <value> • The clientDetails parameter is missing • Invalid request parameter: payKey with value <value> • payKey <value> is expired • Invalid request parameter: <value> with <value> • No sender is associated with this key • The funding plan data is not present • payKey <value> has already been used to make a payment

Code	Message
580022	<ul style="list-style-type: none"> • Invalid request parameter: <value> • Invalid Request Parameter: institutionId with value <value> • The institution id <value> is not activated • The clientDetails parameter is missing • Invalid request parameter: payKey with value <value> • Invalid Phone Country Code <value> • Invalid Country Code <value> for Receiver • Invalid Receiver Phone Type <value> <value> <value> • Invalid Phone Format <value> <value> • Invalid Phone Number <value> <value> • Invalid Receiver Phone Number <value> <value> • Invalid Area or Exchange Code <value> <value> • <value> has already been set and cannot be set multiple times • Invalid request parameter: <value> with value <value> • Invalid request parameter: <value> <value> is invalid • Invalid request parameter: <value> <value> is duplicated • Invalid request parameter: <value> <value> cannot be negative or zero
580022	<ul style="list-style-type: none"> • Invalid request parameter: <value> • Invalid request parameter: payKey with value <value>
580022	<ul style="list-style-type: none"> • Invalid request parameter: <value> • Invalid request parameter: payKey with value <value> • No sender is associated with this key
580022	The funding plan data is not present
580023	<ul style="list-style-type: none"> • The fee payer <value> cannot be used if a primary receiver is specified • The fee payer <value> can only be used if a primary receiver is specified • The fee payer <value> must have exactly 2 receivers • The reverseAllParallelPaymentOnError field can be true only if a primary receiver is not specified • The fee payer <value> cannot be used if a preapproval key is specified • Invalid request: FundingType cannot be specified when PaymentSubType is specified • Your Application is not approved to use the PaymentSubType: <value> • Your Application is not approved to use the PaymentType: <value> • Invalid request parameter: PaymentType should be the same for all receivers <value> • Inconsistent request parameter for digitalgoods • Invalid request parameter: PaymentSubType should be the same for all receivers <value> • Invalid Request: RECEIVABLE cannot be used with other funding types • For chained payments, receivers must have premier or business accounts
580027	<ul style="list-style-type: none"> • Currently the system does not accept the currency <value> • Unsupported country code <value> • Currently the system does not accept the currency <value> • The parameter is not supported
580027	This parameter is not supported

Preface

Revision History

Code	Message
580029	A pin is required to confirm this preapproval Missing required request parameter: <value>
580029	Missing required request parameter: <value>
580029	A pay key, transaction ID, or tracking ID is missing
589018	The pay key <value> is invalid
589019	The preapproval key <value> is invalid
589061	The receiver <value> is invalid for this refund

Revision History

Date	Description
11/15/2010	Revised draft for version 1.6.0. Includes information about preapproved future payments and shipping address information for embedded payments
10/29/2010	Revised draft for version 1.6.0. Includes information about embedded payments and a digital goods example
10/10/2010	Revised draft for version 1.6.0
07/13/2010	Revised draft for version 1.5.0. Includes information about guest payments
06/28/2010	Release draft for version 1.5.0
03/10/2010	Release draft for version 1.4.0
10/09/2009	Release draft for version 1.3.0
09/26/2009	Review draft for version 1.3.0
09/25/2009	Review draft for version 1.3.0
08/27/2009	Revised draft for version 1.3.0
08/17/2009	Revised draft for version 1.2.0.
08/07/2009	Revised draft for version 1.1.0.
07/16/2009	Initial draft for version 1.0.0.

1

Introducing Adaptive Payments

Introduction for the Adaptive Payments API.

The Adaptive Payments API enables you to send money in many different scenarios, from simple to complex; for example, you might build a small send money application for a social networking site or a robust payroll system.

Adaptive Payments API Operations

Adaptive Payments provides API operations, enabling you to build an application that handles payments, preapprovals for payments, refunds, and additional operations related to payments. Some kinds of payments and operations require specific permission to use.

API Operations Related to Payments

API Operation	Description
Pay	Transfers funds from a sender's PayPal account to one or more receivers' PayPal accounts (up to 6 receivers)
PaymentDetails	Obtains information about a payment set up with the Pay API operation
ExecutePayment	Executes a payment
GetPaymentOptions	Obtain the settings specified with the SetPaymentOptions API operation
SetPaymentOptions	Sets payment options

API Operations Related to Preapprovals

API Operation	Description
Preapproval	Sets up preapprovals, which is an approval to make future payments on the sender's behalf
PreapprovalDetails	Obtains information about a preapproval
CancelPreapproval	Cancels a preapproval
ConfirmPreapproval	Confirms that the specified preapproval can be used to make payments

Other API Operations

API Operation	Description
Refund	Refunds all or part of a payment
ConvertCurrency	Obtains the current foreign exchange (FX) rate for a specific amount and currency
GetFundingPlans	Determines the funding sources that are available for a specified payment
GetShippingAddresses	Obtains the selected shipping address
GetAvailableShippingAddresses	Obtains available shipping addresses

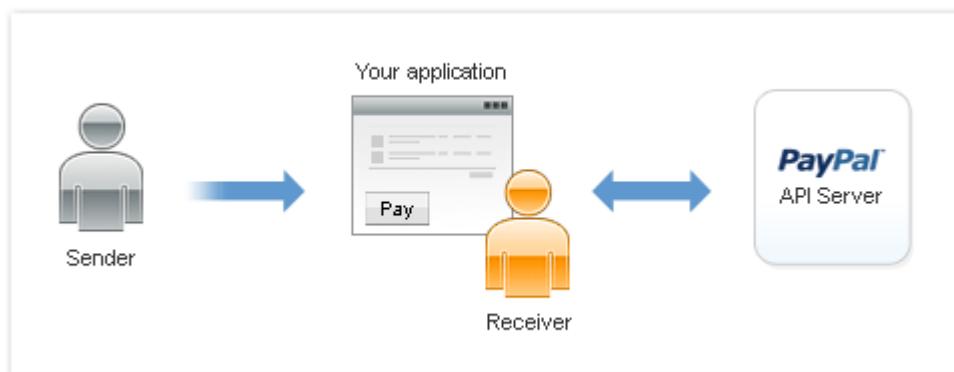
Adaptive Payments Actors and Objects

Adaptive payments handles payments between a *sender* of a payment and one or more *receivers* of the payment. You are an *application owner*, such as a merchant that owns a website, the owner of a widget on a social networking site, the provider of a payment application on cell phones, and so on. Your application is the caller of Adaptive Payments API operations.

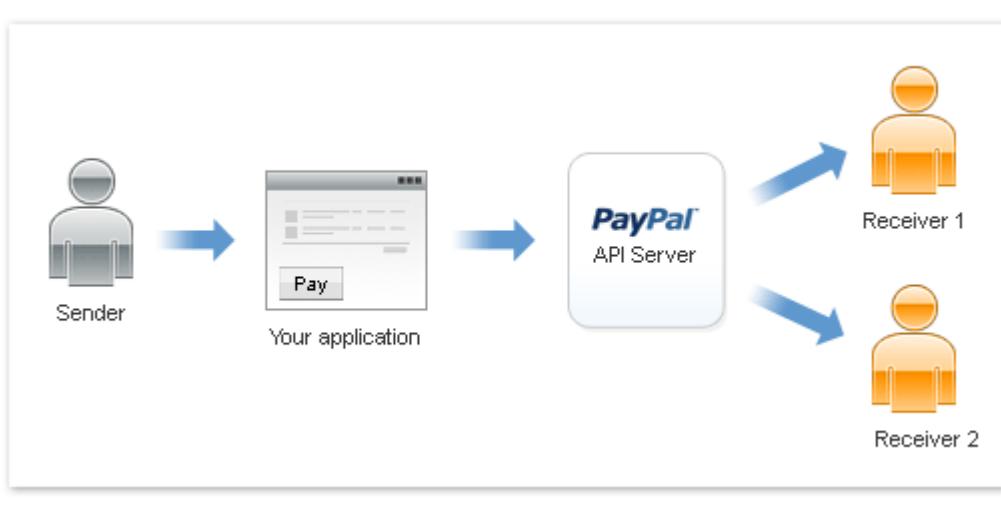
NOTE:

The application owner must have a PayPal business account. Senders and receivers can have any PayPal account type. Senders and receivers are not required to have PayPal accounts initially; however, a sender is prompted to create an account before a payment can be completed and a receiver must create an account to receive the funds after the payment completes.

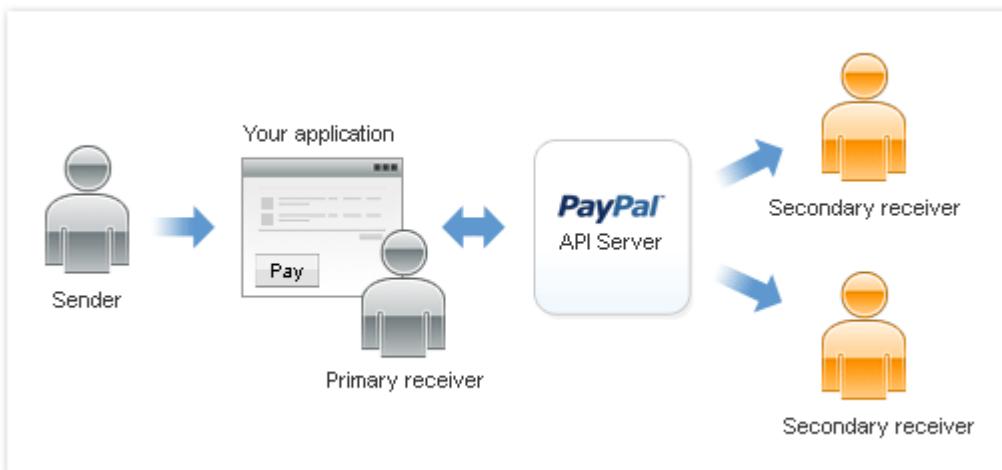
With many applications, you may be both the application owner and a receiver; for example, as the owner of a website, you are the receiver of payments from the senders who are your customers. The following diagram shows the relationship between a sender, you as a receiver, and PayPal:



You are not required to be a receiver. For example, if you own a shopping cart, you are not required to receive payments directly; you can facilitate payments between the sender and receivers that provide the actual goods. The following diagram shows the relationship between a sender, you as an application owner that directs payments to receivers, and PayPal:



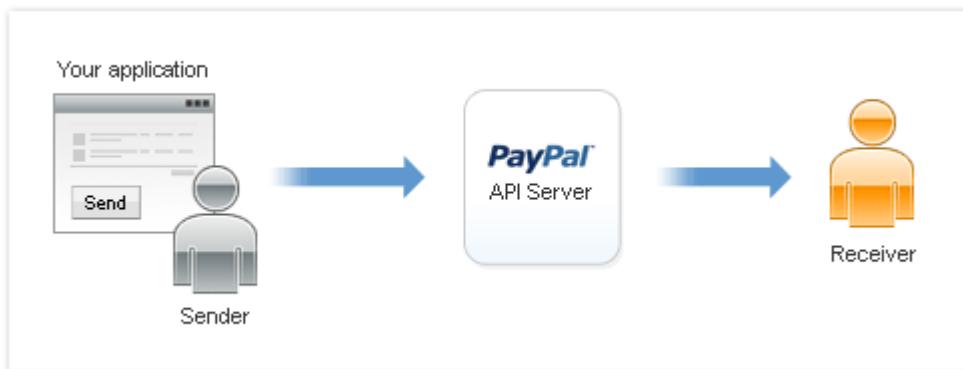
The diagram above shows a payment in which the sender pays multiple receivers in a parallel payment. With parallel payments, the sender can see the transaction to each receiver; this is not the case with chained payments as described below.



The diagram above shows a chained payment, in which your application acts as an intermediary called a primary receiver. The sender pays the primary receiver an amount, which the primary receiver splits between secondary receivers. The sender only sees the transaction to the primary receiver, not the secondary receivers. The secondary receivers only see the primary receiver transaction.

Introducing Adaptive Payments

Simple, Parallel, and Chained Payments



As the diagram above shows, your application can also be the sender. For example, you could use Adaptive Payments to create a sales commission application that transfers funds owed for commissions from your account to the accounts of your sales force.

Simple, Parallel, and Chained Payments

Adaptive Payments provides several methods of payment: simple, parallel, and chained payments. Each method of payment is called with the Pay API; the payment method is determined by how the Pay request is structured.

- Simple payments enable a sender to send a single payment to a single receiver.
This is the traditional way that payments are made. For example, your website could use an Adaptive Payments checkout flow that transfers money from a sale from your customer's PayPal account to your own account.
- Parallel payments enable a sender to send a single payment to multiple receivers.
For example, your application might be a shopping cart that enables a buyer to pay for items from several merchants with one payment. Your shopping cart allocates the payment to merchants that actually provided the items. PayPal then deducts money from the sender's account and deposits it in the receivers' accounts.
- Chained payments enable a sender to send a single payment to a primary receiver; the primary receiver keeps part of the payment and pays secondary receivers the remainder.
For example, your application could be an online travel agency that handles bookings for airfare, hotel reservations, and car rentals. The sender sees only you as the primary receiver. You allocate the payment for your commission and the actual cost of services provided by other receivers. PayPal then deducts money from the sender's account and deposits it in both your account and the secondary receivers' accounts.

NOTE: Chained payments also includes delayed chained payments, in which payments to secondary receivers can be delayed for up to 90 days.

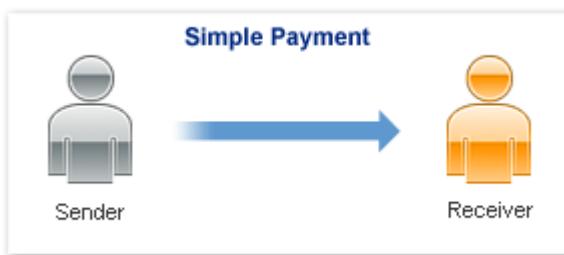
Simple Payments

Simple payments enable a sender to send a single payment to a single receiver. This is sometimes considered a traditional payment, such as a payment from a buyer to a seller.

Scenarios involving simple payments might include the following scenarios:

- Buyer making a payment on a merchant's website.
- Buyer making a single payment for a cart of items from the same merchant.
- Person on a social networking site making a payment for a purchase to the receiver; for example, a sender sends money to pay for lunch at a restaurant.

In these cases, the sender sends a payment to a single receiver. The following example shows a sender making a simple payment:



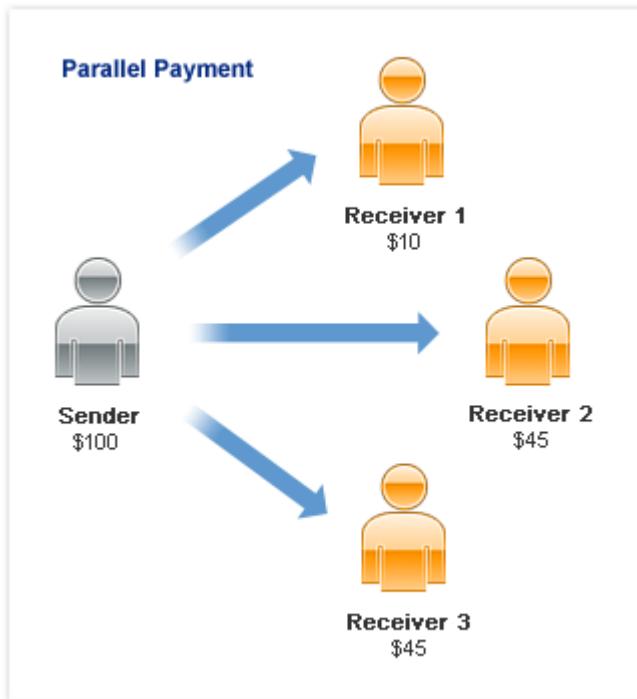
Parallel Payments

A parallel payment is a payment from a sender that is split directly among 2 to 6 receivers. Technically, a parallel payment is a set of multiple payments made in a single Pay request.

Parallel payments are useful in cases when a buyer intends to make a single payment for items from multiple sellers. Examples include the following scenarios:

- a single payment for multiple items from different merchants, such as a combination of items in your inventory and items that are drop-shipped by partners.
- purchases of items related to an event, such as a trip that requires airfare, car rental, and a hotel booking.

In these cases, the sender knows the receivers and the amount paid to each one. The following example shows a sender paying 3 receivers in a single parallel payment:



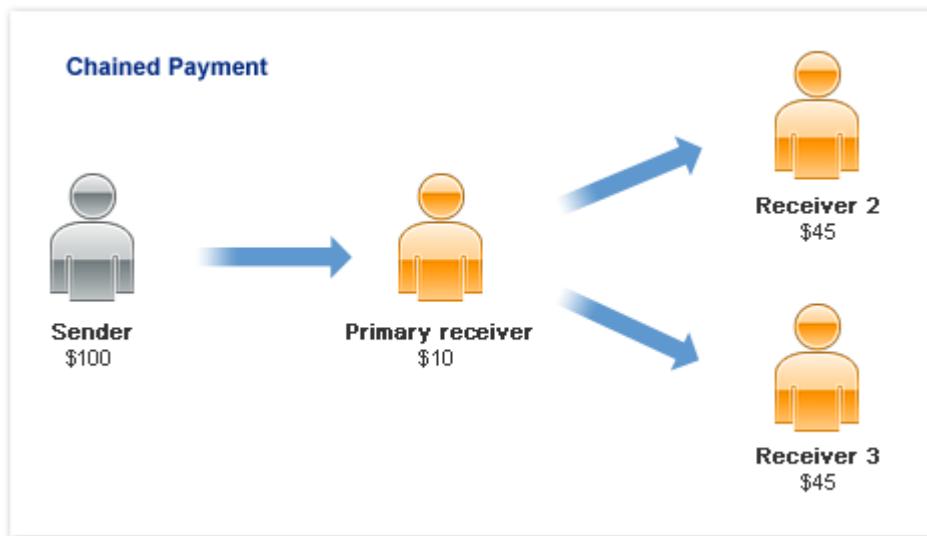
NOTE: The scenario above is an example only and does not take PayPal fees into account.

Chained Payments

A chained payment is a payment from a sender that is indirectly split among multiple receivers. It is an extension of a typical payment from a sender to a receiver; however, a receiver, known as the *primary receiver*, passes part of the payment to other receivers, who are called *secondary receivers*.

NOTE: Chained payments require a specific permission level on the part of the API caller and merchant. For information, refer to the section Adaptive Payments Permission Levels.

You can have at most one primary receiver and from 1 to 5 secondary receivers. Chained payments are useful in cases when the primary receiver acts as an agent for other receivers. The sender deals only with the primary receiver and does not know about the secondary receivers, including how a payment is split among receivers. The following example shows a sender making a payment of \$100:



In this example, the primary receiver receives \$100 from the sender's perspective; however, the primary receiver actually receives only \$10 and passes a total of \$90 to secondary receivers Receiver 2 and Receiver 3.

NOTE: The scenario above is an example only and does not take PayPal fees into account.

Delayed Chained Payments

By default, payments to all receivers in a chained payment are immediate; however, you can choose to delay a payment to a secondary receiver. For example, as primary receiver, you may require secondary receivers to perform some action, such as shipping goods or waiting for expiration of a return period, before making payment. In this case, after the sender pays you, you must explicitly execute a payment to secondary receivers. The payment must occur within 90 days, after which the payment cannot be made as part of the original chained payment.

Payment Approval

The sender of a payment must approve the transfer. The sender can log in to paypal.com to approve each payment, preapprove payments, or when the sender is your application, be implicitly approved to make payments.

NOTE: Preapproval requests require an advanced permission level. For information, refer to the Adaptive Payments Permission Levels section.

There are 3 kinds of payment approvals:

- Explicit approval payments, in which the sender logs into paypal.com to approve each payment.

Explicitly approving payments is the traditional way to pay with PayPal. This method is the only option unless the sender has set up a preapproval agreement or you, the API caller, are also the sender.

Introducing Adaptive Payments

Adaptive Payments Flows

- Preapproved payments, in which a sender logs into PayPal and sets up preapprovals that approve future payments; for example, for a specific vendor.
The sender logs into paypal.com once to set up the preapproval. After the preapproval is set up, payments are considered approved, and specific approval is unnecessary.
- Implicit approval payments, in which your application is both the sender of a payment and the caller of the Adaptive Payments Pay API.
In this case, the payment is drawn from your own account, which eliminates the need for approval.

Adaptive Payments Flows

Adaptive Payment flows are largely determined by the type of approval necessary for a payment.

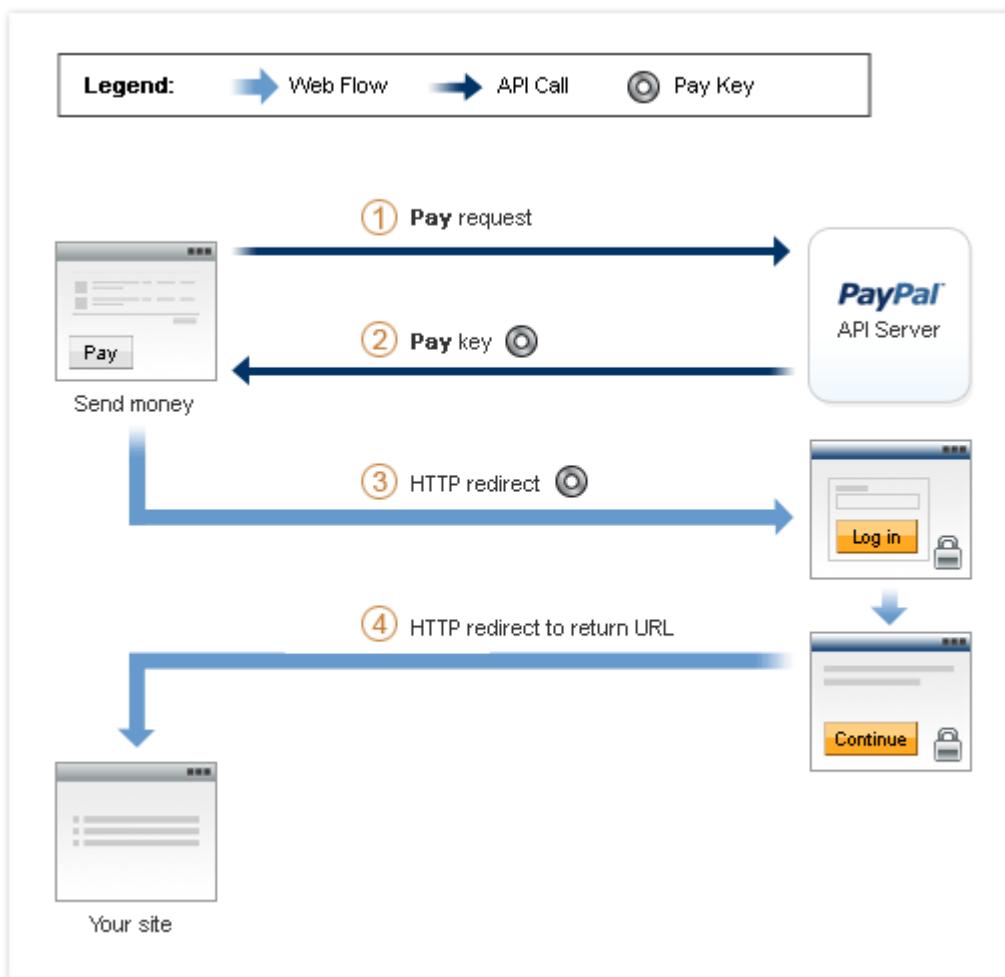
Payments can be approved explicitly, implicitly, or preapproved as described in the Payment Approval section. The type of approval determines how your application interacts with Adaptive Payments payment flows.

Explicitly approved payments require the sender to log in to paypal.com. Preapproved and implicitly approved payments are approved follow a different flow.

Explicit Approval Payments Flow

Explicit approval payments require the sender to log in to paypal.com to approve the payment. You control the interaction between your application and PayPal by specifying URLs for redirection in various situations.

The following diagram shows the basic flow of control for a payment (such as payment for items in a shopping cart at a web store) that requires the sender to log in to paypal.com to approve the payment:



The following items correspond to the circled numbers in the diagram:

1. Your site or device sends a Pay request to PayPal on behalf of a sender.
2. PayPal responds with a key that you use when you direct the sender to PayPal.
3. You redirect your sender's browser to paypal.com.
4. After the sender authorizes the transfer of funds, PayPal redirects your sender's browser to the location you specify.

NOTE: The cancel operation is not shown in the above diagram. Cancellation is handled by a separate cancellation URL to which PayPal redirects the sender's browser any time the sender cancels while on paypal.com.

In addition to the steps shown above, PayPal sends email to you and the sender when a payment is made.

Introducing Adaptive Payments

Adaptive Payments Flows

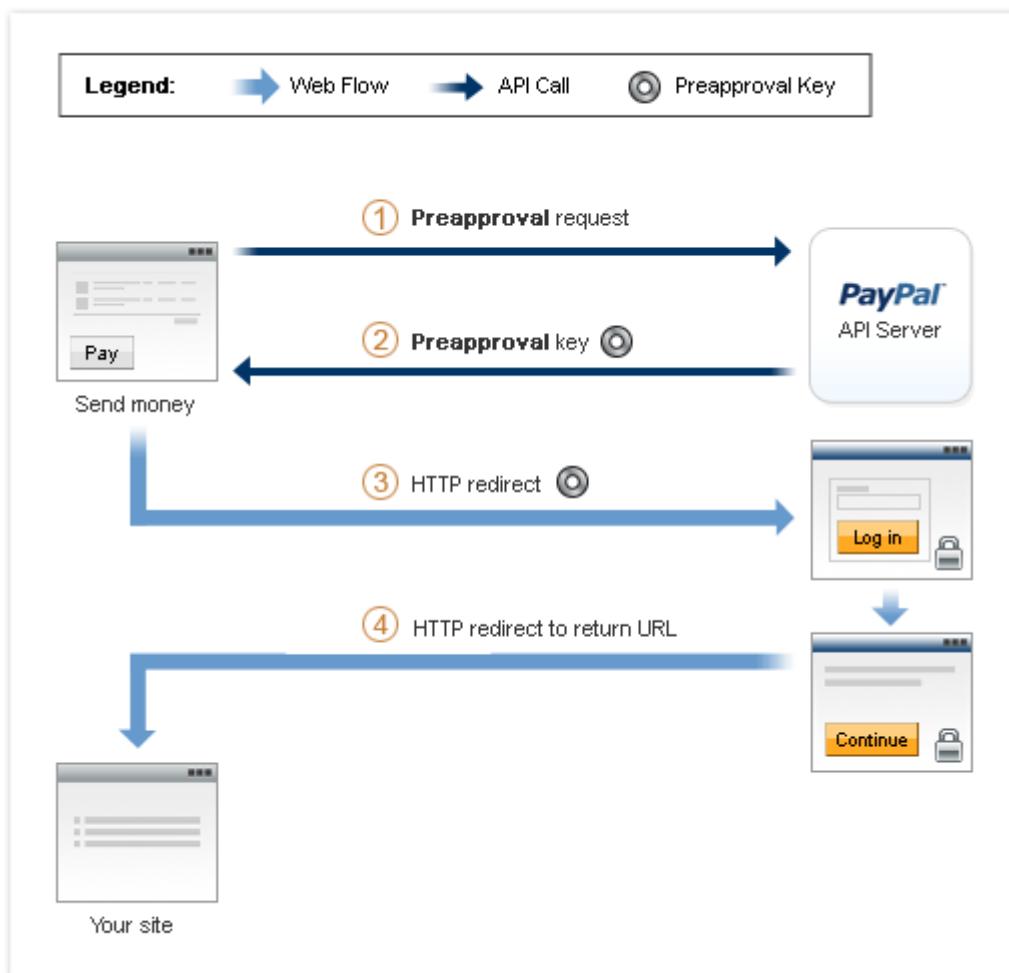
Preapproved Payments Flow

Preapproved payments require the sender to log in to paypal.com to set up the payment agreement with a particular vendor. You control the interaction between your application and PayPal by specifying URLs for redirection in various situations.

The sender logs in to paypal.com and sets up the preapproval, which includes setting the following information:

- duration of the preapproval, from the start date to the end date, inclusive
- the maximum amount being preapproved
- the maximum number of payments

The following diagram shows the basic flow of control during a preapproval operation:



The following items correspond to the circled numbers in the diagram:

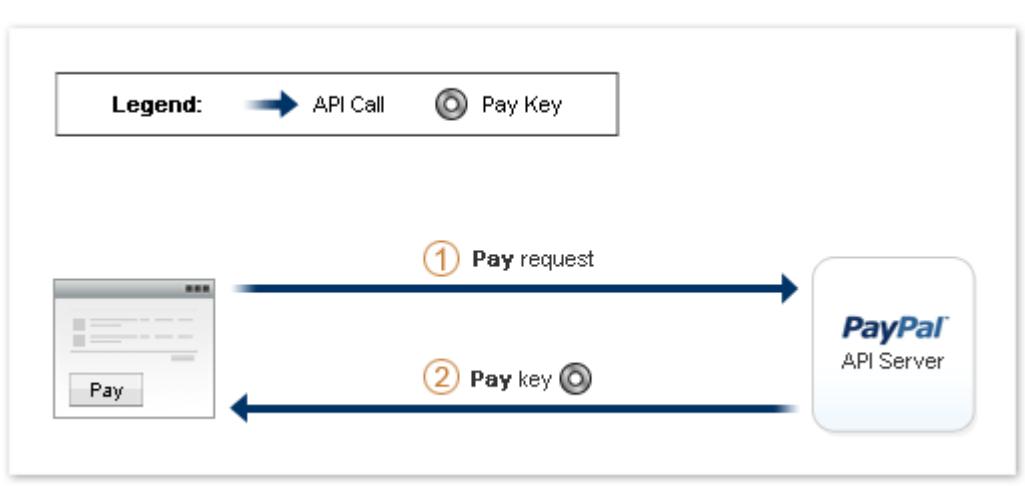
1. Your site or device sends a Preapproval request to PayPal on behalf of a sender.

2. PayPal responds with a key, called a *preapproval key*, that you use when you direct the sender to PayPal, and once the preapproval has been established, whenever you automatically complete a payment on behalf of the sender.
3. You redirect your sender's browser to PayPal.
4. After the sender logs in to paypal.com and sets up the preapproval, PayPal redirects the sender's browser to a location you specify.

NOTE: The cancel operation is not shown in the above diagram. Cancellation is handled by a separate cancellation URL to which PayPal redirects the sender's browser any time the sender cancels while on paypal.com.

In addition to the steps shown above, PayPal sends an email notifying you and the sender that the preapproval has been created.

After the sender sets up the approval, you can make payments on the sender's behalf directly. The sender is no longer required to log in to PayPal to complete the payment. The following diagram shows the basic flow of control during a Pay operation:



The following items correspond to the circled numbers in the diagram:

1. Your site or device sends a Pay request to PayPal on behalf of a sender. You can require the sender to provide a personal identification number (PIN); however, logging in to paypal.com is no longer required.

NOTE: You must provide a preapproval key that identifies the agreement.

2. PayPal still responds with a payment key that you can use for other API operations, such as for obtaining details of the payment or for issuing a refund.

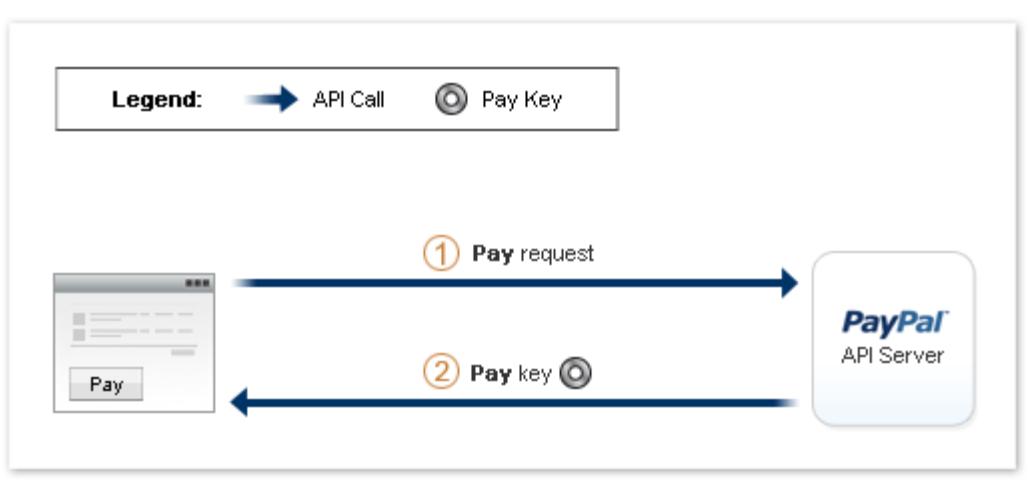
Implicit Approval Payments Flow

Implicit approval payments are payments where the sender and the API caller are using the same account. Because the funds for the payment are drawn from your own account, there is no approval necessary, and as such there is no visible flow for implicit approval payments.

Introducing Adaptive Payments

Adaptive Payments Service Permissions

The following diagram shows the basic flow of control during an implicitly approved payment operation:



The following items correspond to the circled numbers in the diagram:

1. Your site or device sends a Pay request to PayPal.
NOTE: A web flow is not required.
2. PayPal responds with a key that you use for other API operations.

Adaptive Payments Service Permissions

Adaptive Payments services are divided into two categories: *standard services* that do not require specific permission to use and *advanced services* that require permission from PayPal to use. You obtain permission to use a service when you submit an application to PayPal.

You can use the following standard services without requesting specific permission:

- Making simple or parallel payments that require explicit approval of the sender
- Getting payment details
- Making refunds
- Performing currency conversions

To use any other service, you must receive permission from PayPal to use the service when you submit your application. For example, if your application makes chained payments, which is an advanced service, you request permission when you submit your application. Later, if you modify your application to support preapprovals, which is also an advanced service, you will need to resubmit your application to obtain permission from PayPal to use the service.

IMPORTANT: If you allow your application to be executed by a third party to PayPal on your behalf, the third party becomes the API caller because the party is now calling the Adaptive Payments APIs. The third party must also have permission from

PayPal to use the advanced service. For example, if an application supports chained payments, both you and the third party must have permission to use the service.

Embedded Payments

An *embedded payment* is a payment that initiates a visual presentation of the Adaptive Payments payment flow in which the sender appears to never leave your checkout or payment page. Embedded payments make it easier for a sender to make a payment because PayPal may allow the sender to bypass the PayPal login step.

The ability to bypass the login relies on a *remember me* cookie, which is stored in the sender's browser when the sender chooses to take advantage of being remembered. Opting in reduces the number of steps to purchase goods or services without significantly increasing the risk that the sender's account might be misused. After the initial login, PayPal bypasses the login step if subsequent payments meet specific criteria, such as a subsequent payment for a small amount.

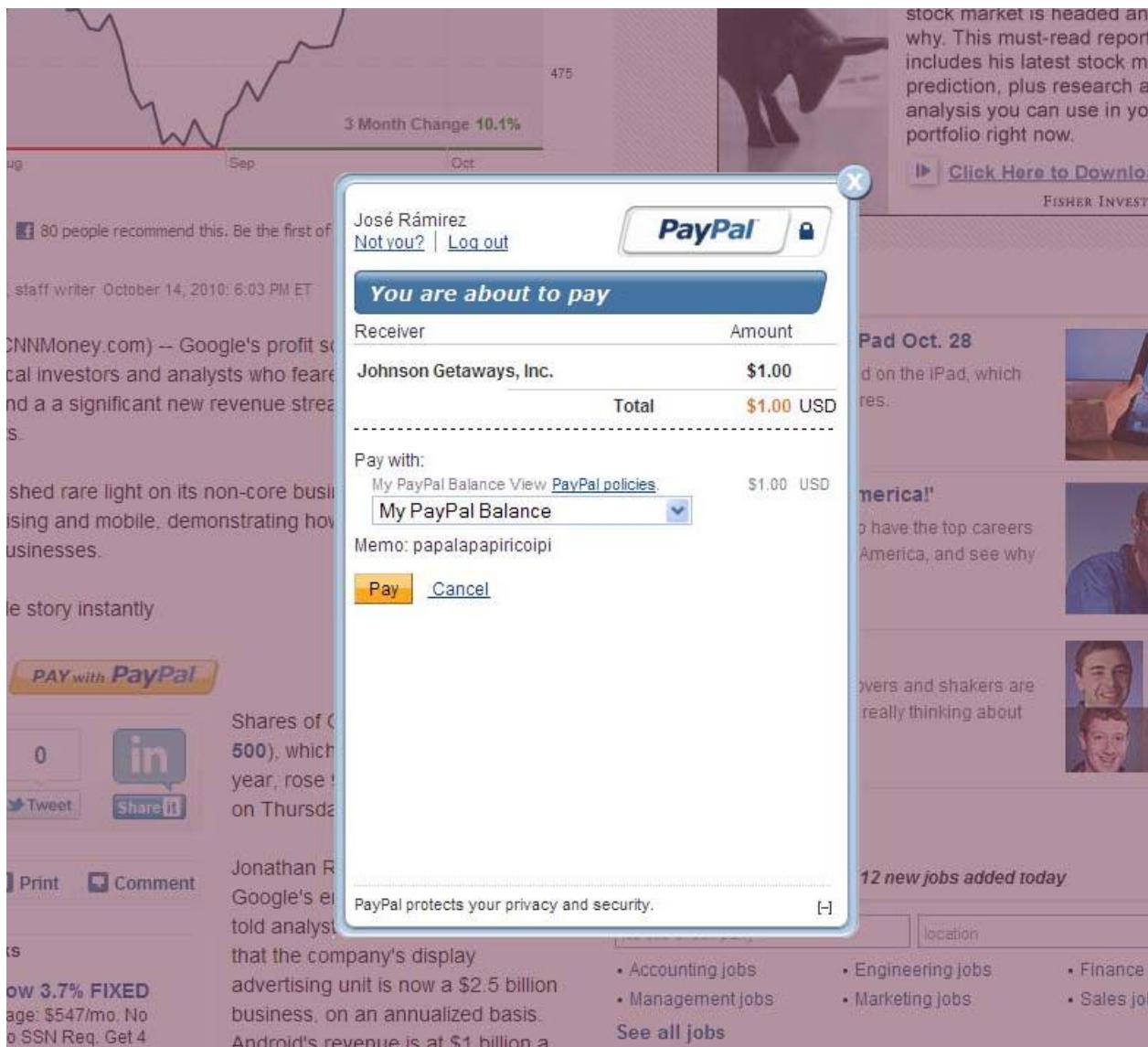
Embedded Payment Flow Presentations

PayPal provides two kinds of visual presentations for the embedded payment flow:

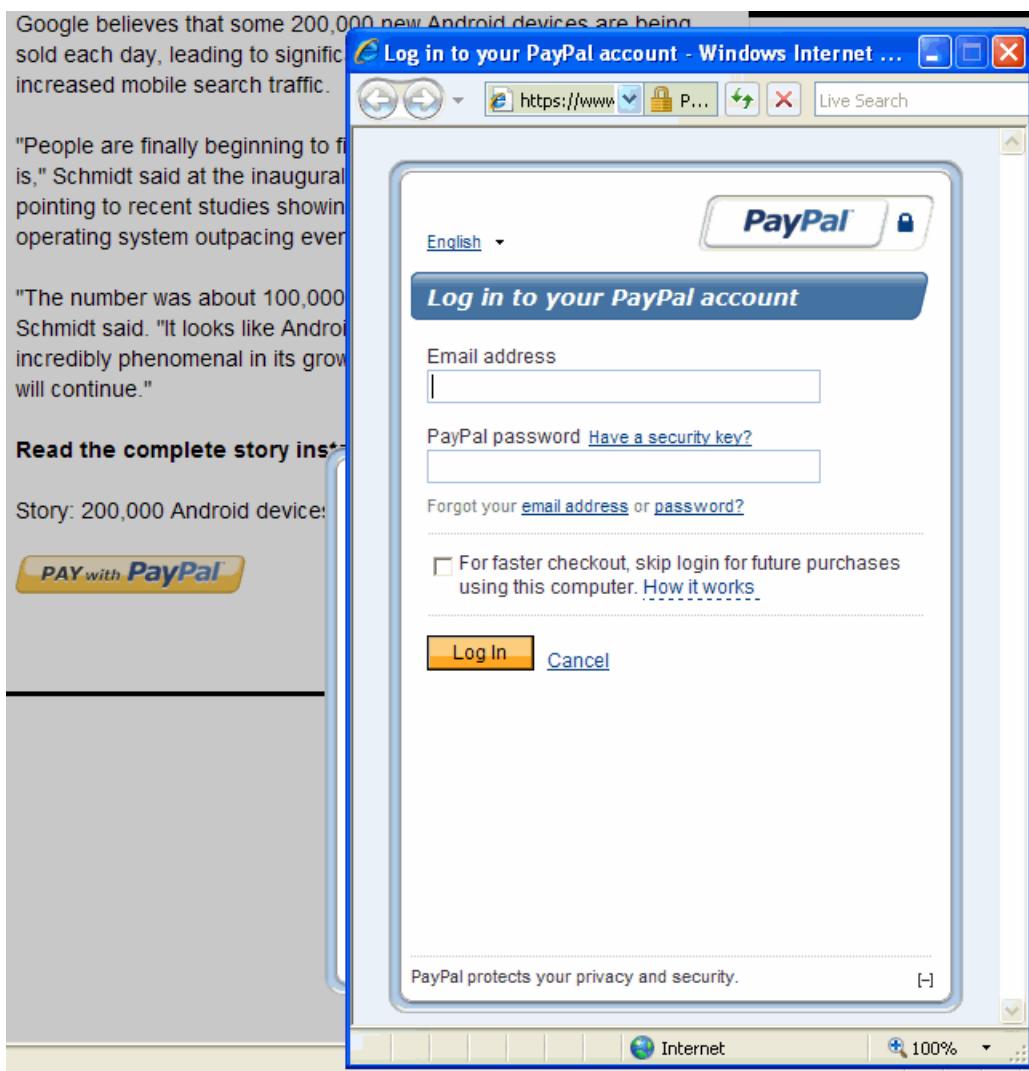
- The payment flow can be embedded as a lightbox on your web page, which gives the impression that the payment is handled completely within your website:

Introducing Adaptive Payments

Embedded Payments



- The payment flow can appear in a minibrowser window that appears in front of your web page, which allows the minibrowser to be moved to better see the contents of your page:



You choose your preferred visual presentation when you invoke the embedded payment flow. In some cases, PayPal may override your choice; for example, when the sender is required to log into PayPal for the initial payment.

Kinds of Embedded Payments

Embedded payments can include

- simple payments
- parallel payments
- chained payments

You can also associate a preapproval for future payments or specify a shipping address with an embedded payment.

Introducing Adaptive Payments

Embedded Payments

IMPORTANT: Payments for digital goods are always embedded payments and must use the embedded payment flow. You cannot associate a preapproval for future payments with a payment for digital goods.

Embedded Payments Implementation Basics

To implement the embedded payment flow, you must

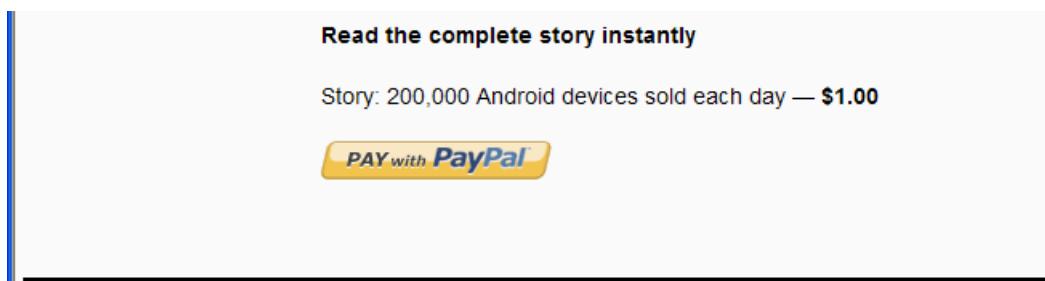
- include JavaScript code from PayPal on your checkout or payment web pages
- use the functions provided in the JavaScript to coordinate the PayPal flow with the appearance of your web pages
- launch your preferred embedded payment flow and redirect the sender's browser to the PayPal URL that supports embedded payments, which is
`https://paypal.com/webapps/adaptivepayment/flow/pay?paykey=...`

You must call the Pay API operation to obtain a payment key before launching the embedded payment flow. If the payment is specifically for digital goods, modify your Pay API operation to specify that each receiver is receiving payment for digital goods.

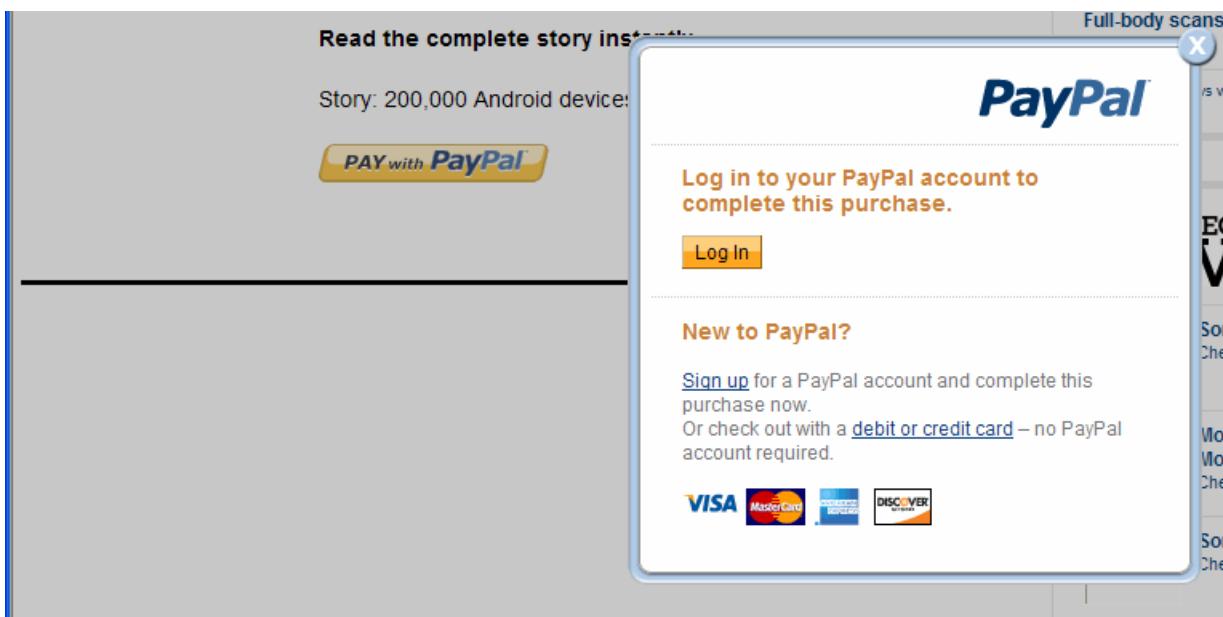
Embedded Payment Experience

To the sender of a payment, the embedded payment experience appears to be built into your website. The PayPal-supplied JavaScript provides all the code needed to set up the flow as an IFRAME within the sender's browser and as a pop-up mini-browser that appears in front of your website.

Typically, the sender initiates a payment by clicking a button:



PayPal responds to the JavaScript that initiates the flow. If it is the first payment, or if PayPal determines that the payment requires the sender to log in, PayPal displays a **Log In** button in the IFRAME created by the JavaScript:



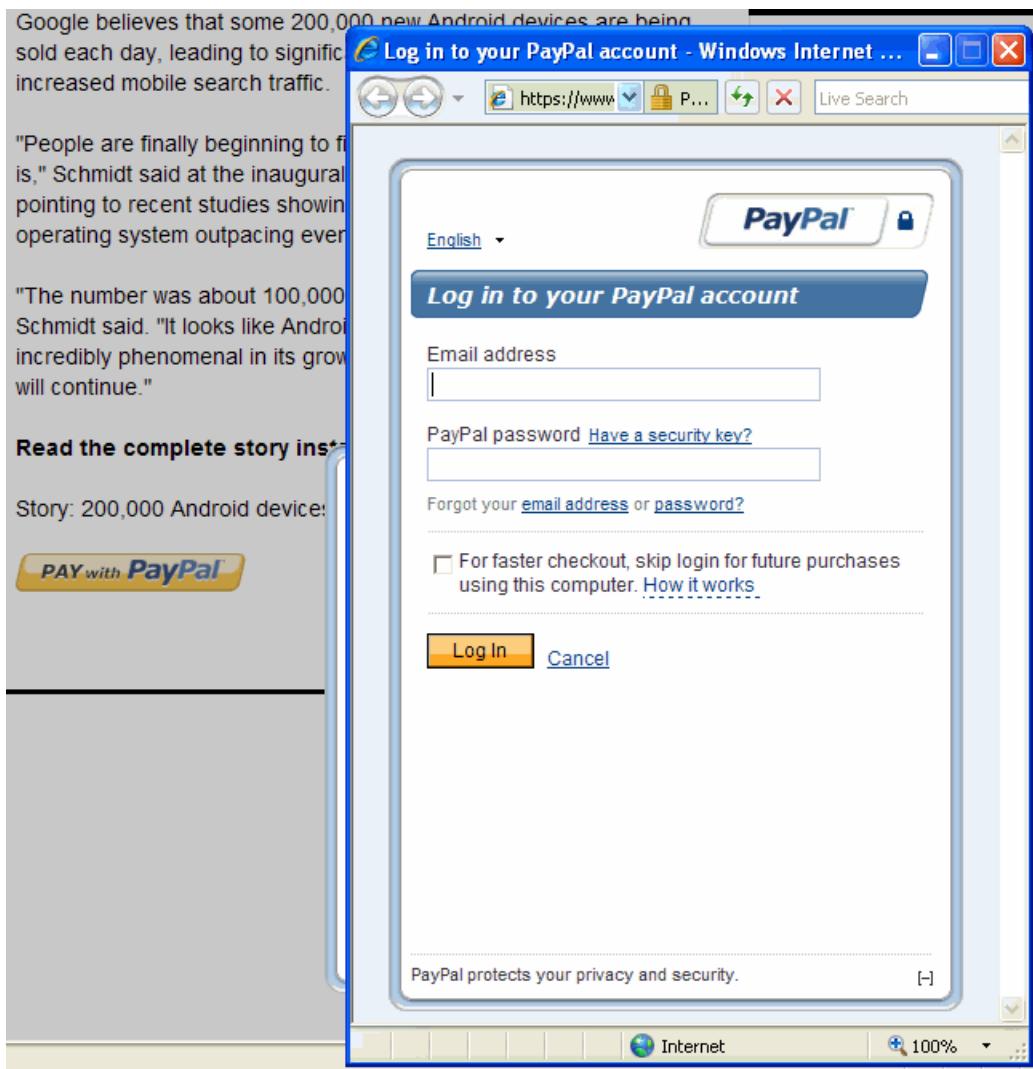
The IFRAME also allows the sender to sign up for a PayPal account or pay as a guest without logging in.

NOTE: Guest checkout only provides the visual benefits of an embedded payment. It does not reduce the payment steps.

After clicking Log In, PayPal displays a **Log in to your PayPal account** page in the minibrowser. The sender enters an email address and password and can also check a box, which allows the sender to skip this step for subsequent purchases:

Introducing Adaptive Payments

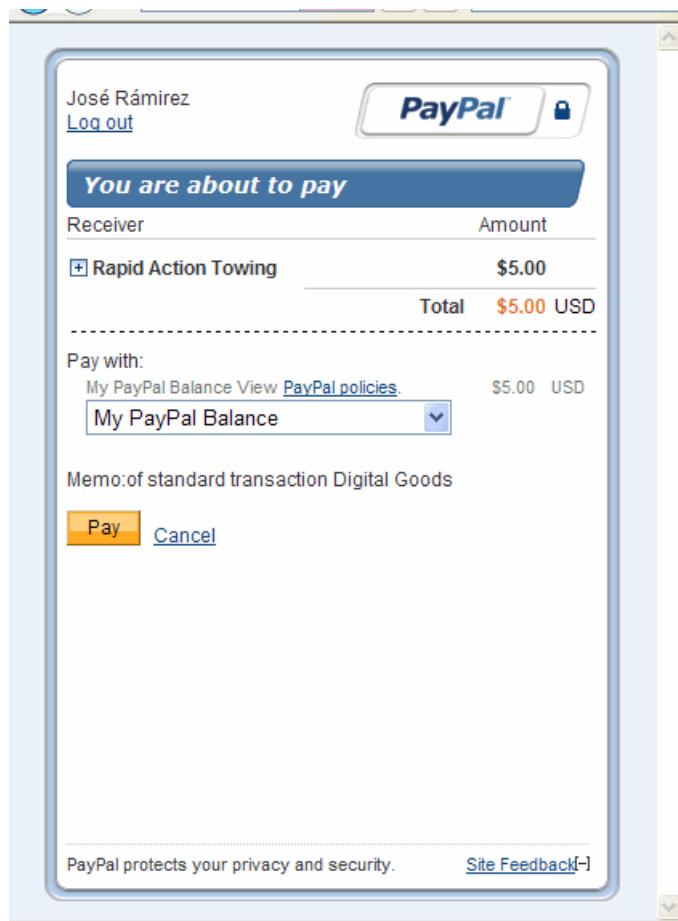
Embedded Payments



The checkbox controls the *remember me* behavior for log in. This behavior allows the sender to skip the log in step in cases where there is little risk of the account being misused.

IMPORTANT: Opting in to the remember me behavior does not guarantee that the sender will not be asked to provide log in credentials.

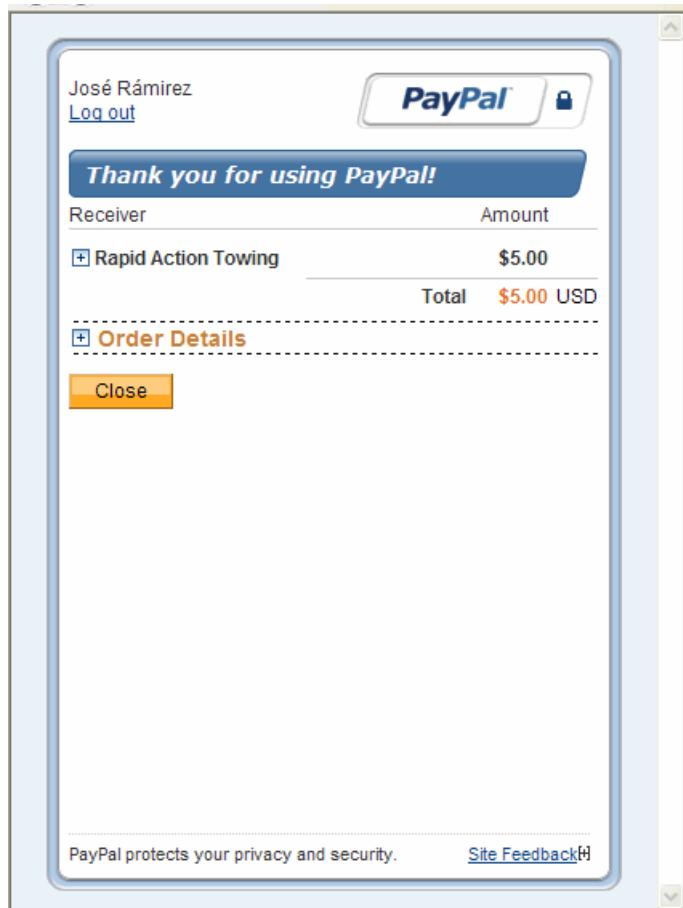
After logging in, PayPal displays the **You are about to pay** page, sometimes called the *review your payment* page in the minibrowser:



If the sender chooses **Cancel**, PayPal redirects the sender's browser to the cancel URL specified in the Pay API operation's request message. If the sender chooses **Pay**, the **Thank you for using PayPal** page appears in the minibrowser:

Introducing Adaptive Payments

Embedded Payments

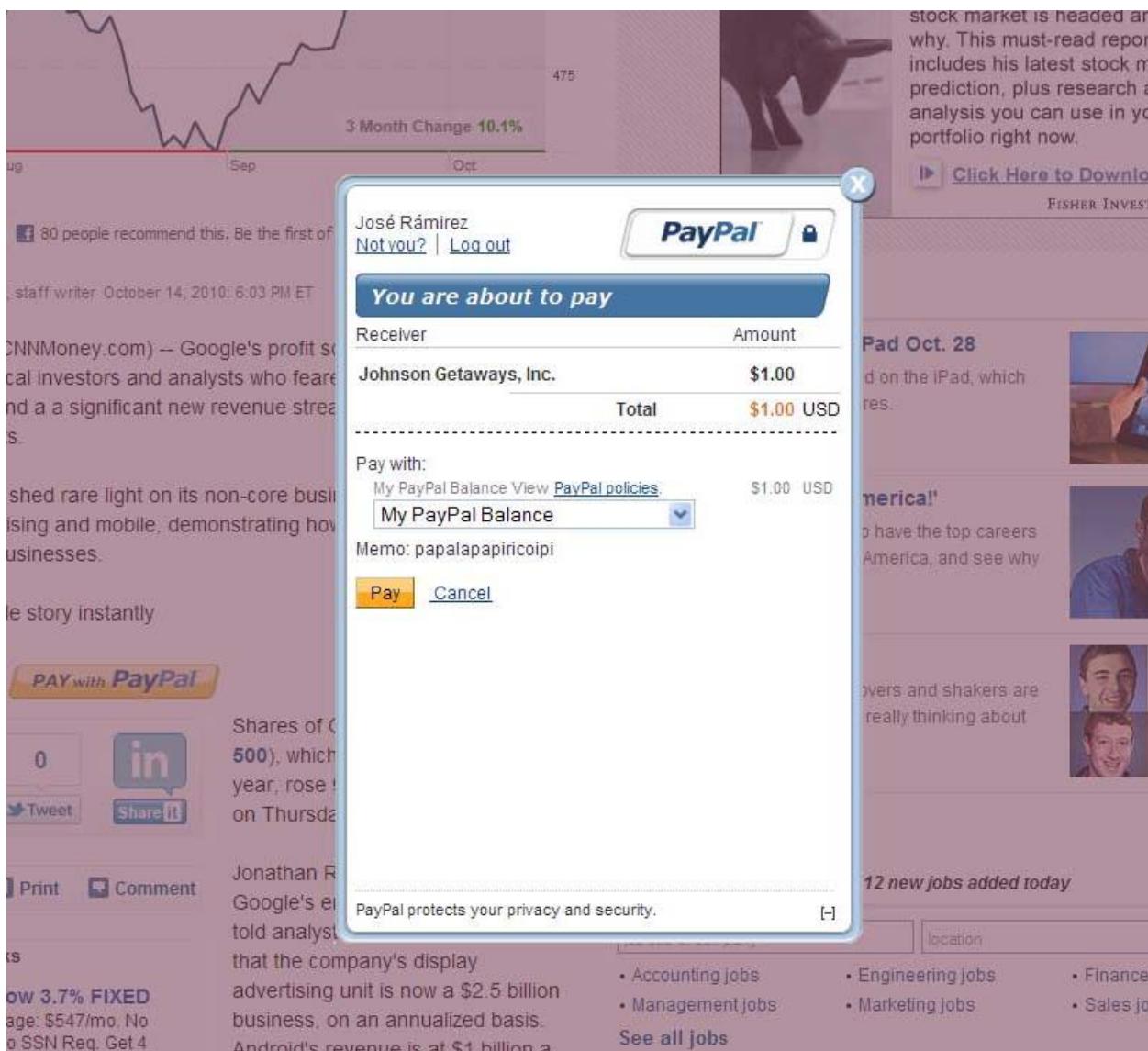


When the sender clicks **Close**, PayPal redirects the sender's browser to the return URL specified in the Pay API operation's request message.

NOTE: You are responsible for closing the minibrowser after PayPal redirects to the page specified in either the return or cancel URL. PayPal provides a JavaScript function that you call to close a PayPal minibrowser or lightbox.

For subsequent payments in which PayPal does not require the account holder to log in again and the account holder has chosen to be remembered, the PayPal payment pages appear in a lightbox rather than in a minibrowser and PayPal bypasses the log in steps. For these payments, the actions you take to launch the flow and close the lightbox, remain the same.

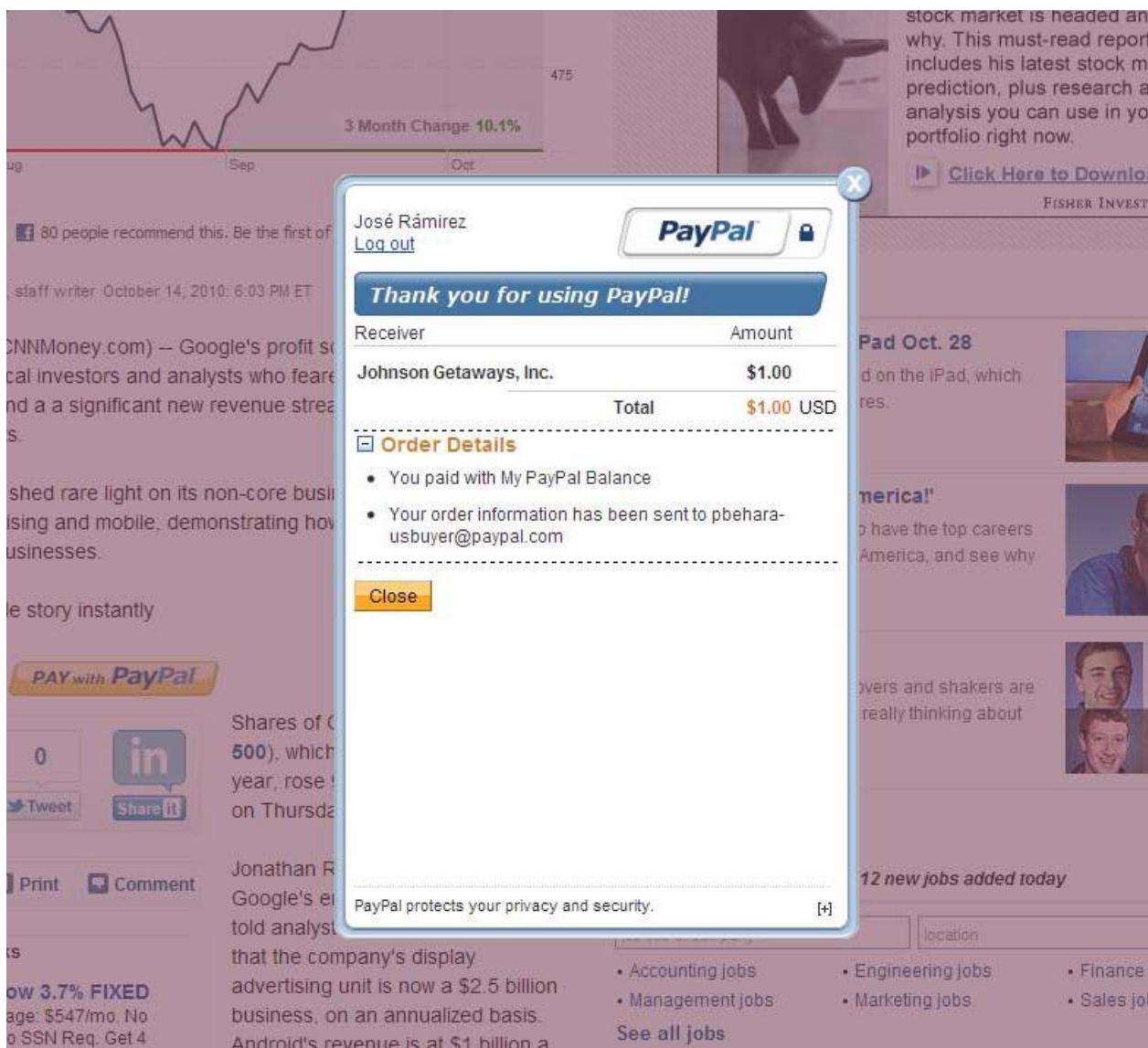
For example, the lightbox for reviewing a payment would appear in your page as follows:



Thelightbox containing the confirmation would appear in your page as follows:

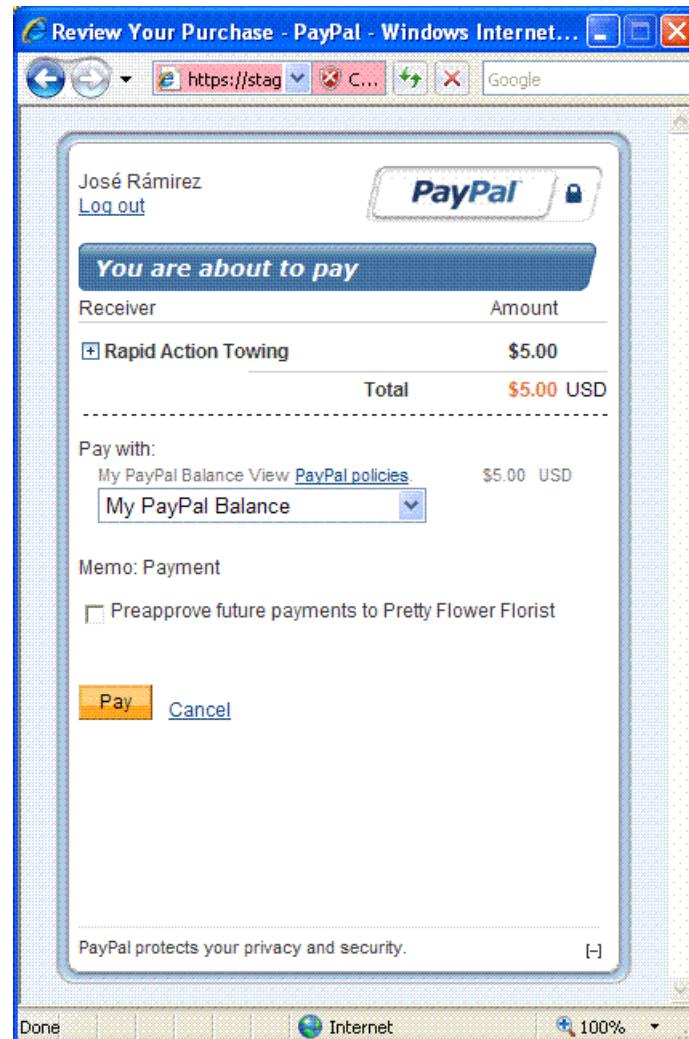
Introducing Adaptive Payments

Embedded Payments



Preapprove Future Payments Checkbox

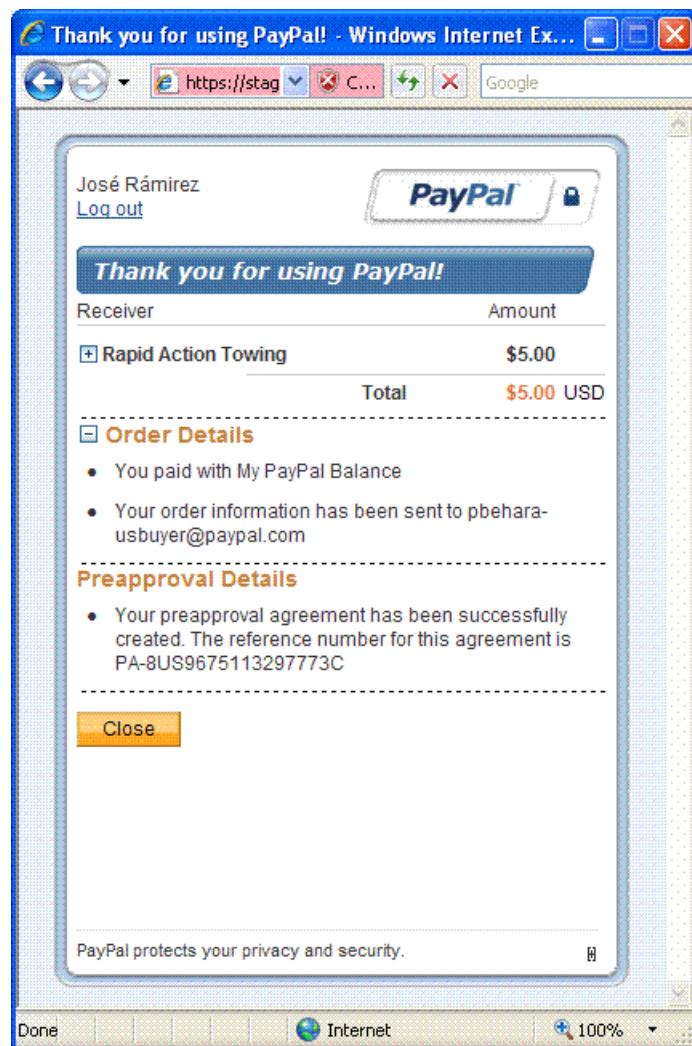
You can add a **Preapprove future payments** checkbox to the sender's embedded payment experience, which enables the sender to preapprove subsequent payments. If you specify both a payment key and a preapproval key when you call the Pay API operation, PayPal displays the checkbox on the payment page in the minibrowser:



If the payment sender checks the preapproval box, the confirmation page provides details about the preapproval:

Introducing Adaptive Payments

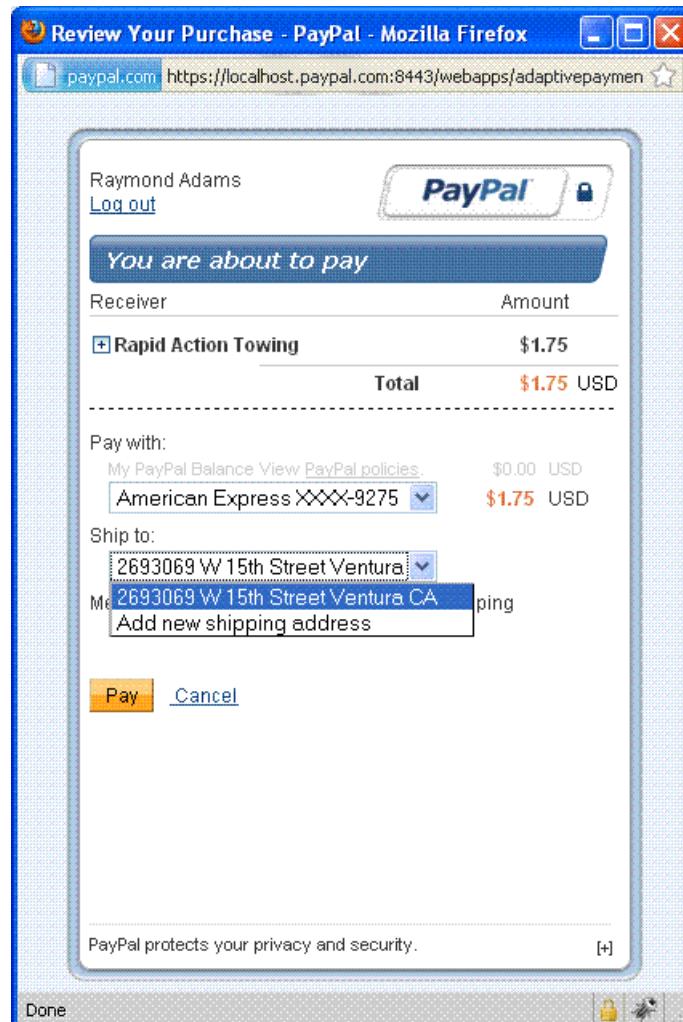
Embedded Payments



NOTE: Unless there is an error with the payment itself, PayPal transfers the money regardless of whether the preapproval for future payments was successful.

Shipping Address Information

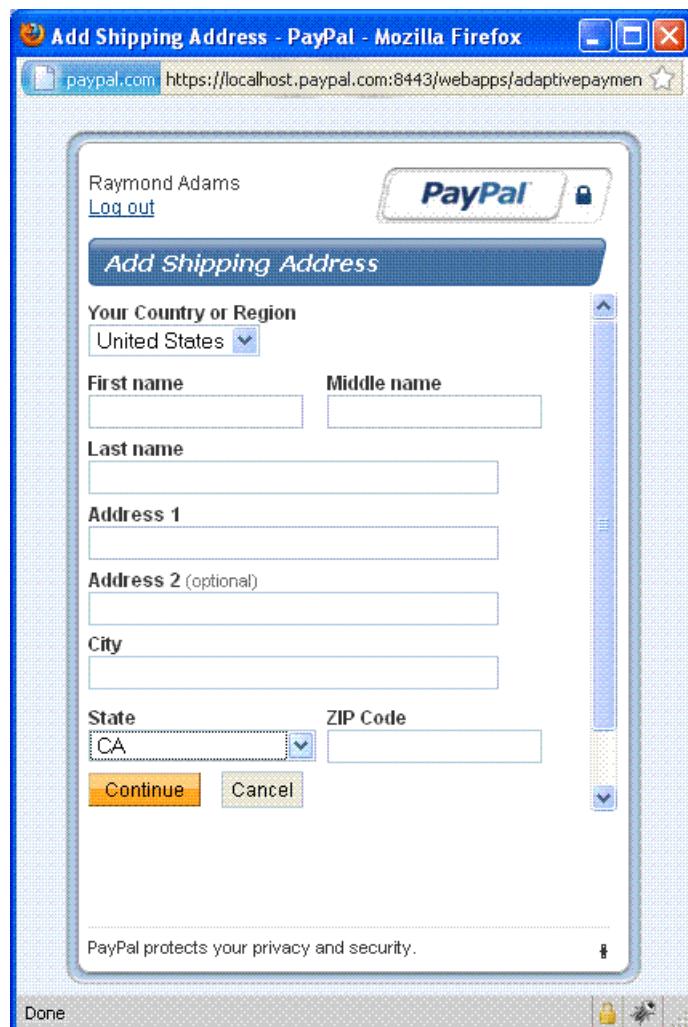
You can display and collect shipping address information. PayPal displays the default shipping address when you set `senderOptions.requireShippingAddressSelection` to true in your request to `SetPaymentOptions`:



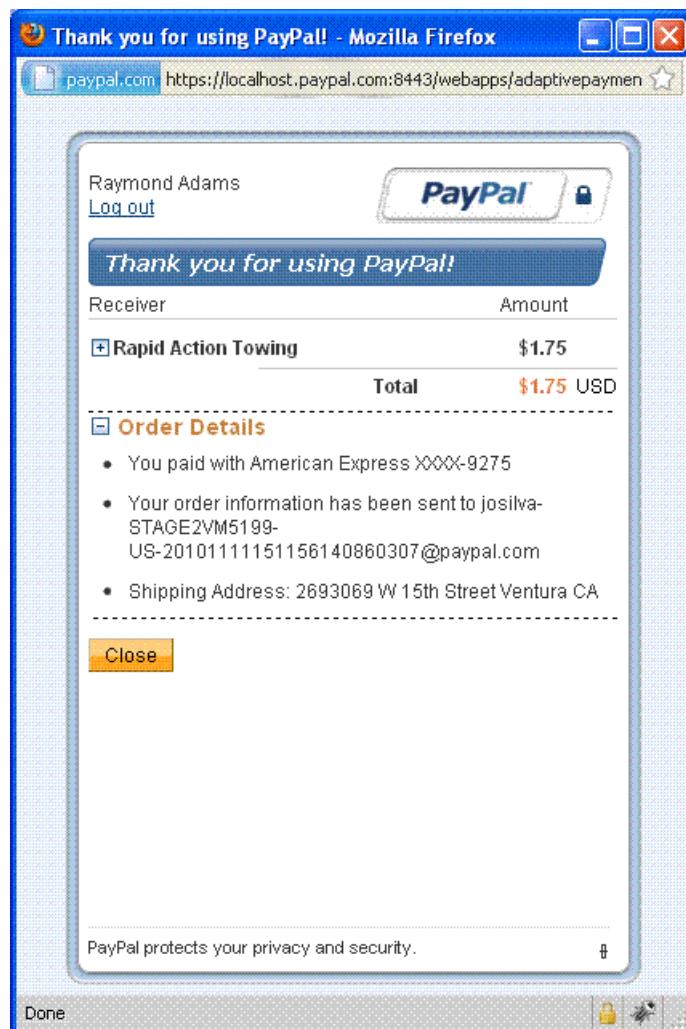
The sender of a payment can select one of the available shipping addresses or add a new shipping address by selecting **Add new shipping address** from the **Ship to:** drop-down menu:

Introducing Adaptive Payments

Embedded Payments



After the sender of the payment clicks **Pay**, PayPal displays the selected shipping address on the **Thank you for using PayPal** page:



Setting Up Web Pages to Invoke the Embedded Payment Flow

Use the JavaScript functions in

<https://www.paypalobjects.com/js/external/dg.js> to set up and control the embedded payment flow. This example shows how to initiate the embedded payment flow after obtaining a payment key.

This example assumes that you obtain a payment key before initiating the flow and that the key does not change or expire before the sender completes the flow.

To set up a web page to invoke the embedded payment flow:

1. Call the Pay API operation to obtain a valid pay key.
2. Create your form or button.

- You must include the pay key and redirect to <https://paypal.com/webapps/adaptivepayment/flow/pay>.
- Optionally, include a preapproval key if you want to enable the selection of **Preapproval for future payments**
- Specify that a lightbox opens in the PayPal-created IFRAME, PPDGFrame.
- Set the expType parameter to indicate your preference for the context in which the PayPal payment flow is displayed. Specify light for lightbox or mini for minbrowser.

```
<form action=
      "https://paypal.com/webapps/adaptivepayment/flow/pay"
      target="PPDGFrame">
  <input id="type" type="hidden" name="expType" value="light">
  <input id="paykey" type="hidden" name="paykey" value="AP-...">
  <input id="preapprovalkey"
         type="hidden" name="preapprovalkey" value="PA-...">
  <input type="submit" id="submitBtn" value="Pay with PayPal">
</form>
```

NOTE: To modify an existing application to use the embedded payment flow, change the redirect from https://www.paypal.com/webscr?cmd=_ap-payment&paykey=... to <https://paypal.com/webapps/adaptivepayment/flow/pay?paykey=...> after obtaining the pay key.

3. Include the PayPal JavaScript functions from dg.js.

```
<script src="https://www.paypalobjects.com/js/external/dg.js">
</script>
```

4. Create an embedded flow object and associate it with your payment form or button.

```
<script>
var dgFlow = new PAYPAL.apps.DGFlow({ trigger: 'submitBtn' });
</script>
```

After Completing This Task:

On the pages you identify as the return and cancel URLs in the Pay API operation, you must include the PayPal JavaScript functions from dg.js and close the PayPal window, as in the following example:

```
dgFlow = top.dgFlow || top.opener.top.dgFlow;
dgFlow.closeFlow();
top.close();
```

Displaying and Collecting Shipping Addresses

PayPal displays the default shipping address and allows the payment sender to change the address when you set `senderOptions.requireShippingAddressSelection` to `true` in your request to the `SetPaymentOptions` API operation. You call the `GetShippingAddresses` API operation to obtain the selected shipping address after invoking the embedded payment flow.

The steps in this example assume that you have implemented the JavaScript for invoking the embedded payment flow, that you have set up your button or form to invoke the flow, and that you have included the code to close the window associated with the flow.

To display and collect the selected shipping address

1. Call the Pay API operation with `actionType` set to `CREATE` to obtain a payment key.
2. Set `senderOptions.requireShippingAddressSelection` to `true` in your request to `SetPaymentOptions` and call the API operation.
3. Redirect the payment sender's browser to the embedded payment flow at <https://paypal.com/webapps/adaptivepayment/flow/pay?paykey=...> after obtaining the pay key.
4. After returning from the flow, call the `GetShippingAddresses` API operation to obtain the selected shipping address.

Guest Payments

Adaptive payments supports *guest payments*, which are payments using a sender's credit card without logging into PayPal to complete the transaction. The sender is not explicitly identified as a PayPal account holder during the transaction and is not required to have a PayPal account.

Each receiver of a guest payment must be a verified PayPal business or premier account holder.

PayPal handles guest payments in the same way that it handles explicitly approved payments. Instead of logging into PayPal to complete transaction, the sender provides credit card information on the PayPal payment screen:

Introducing Adaptive Payments

Fee Payment Configuration

Mid-Valley Construction

Your payment summary

Description	Amount
Friends 'n Stitches Needlework	\$2.00 USD
Raymond Adams	\$3.00 USD
Total:	\$5.00 USD

Choose a way to pay

► Pay with my PayPal account

▼ Pay with a credit or debit card
If you don't have a PayPal account

Country: United States

Credit card number:

Payment types:

Expiration date: mm / yy

CSC:
What is this?

First Name:

Last Name:

Address Line 1:

Address Line 2 (optional):

ZIP Code:

City/State:

Phone:

Create a PayPal account (optional)
It's easy and free to sign up. Learn more [about PayPal](#)

Email address: You'll use this email address to log in to PayPal.

Password: 8 characters minimum

Reenter password:

Secured checkout by

Cancel and return to [Mid-Valley Construction](#)

PayPal. The safer, easier way to pay. For more information, read our [User Agreement](#) and [Privacy Policy](#).

NOTE: For European Union countries, only 10 guest payments are allowed per card.

Fee Payment Configuration

You can set up a payment transaction so that either the sender of a payment pays the fee or the receivers of a payment pay the fee. If receivers pay the fee, you can specify whether the

primary receiver in a chained payment pays the entire fee or whether all receivers pay a portion of the fee.

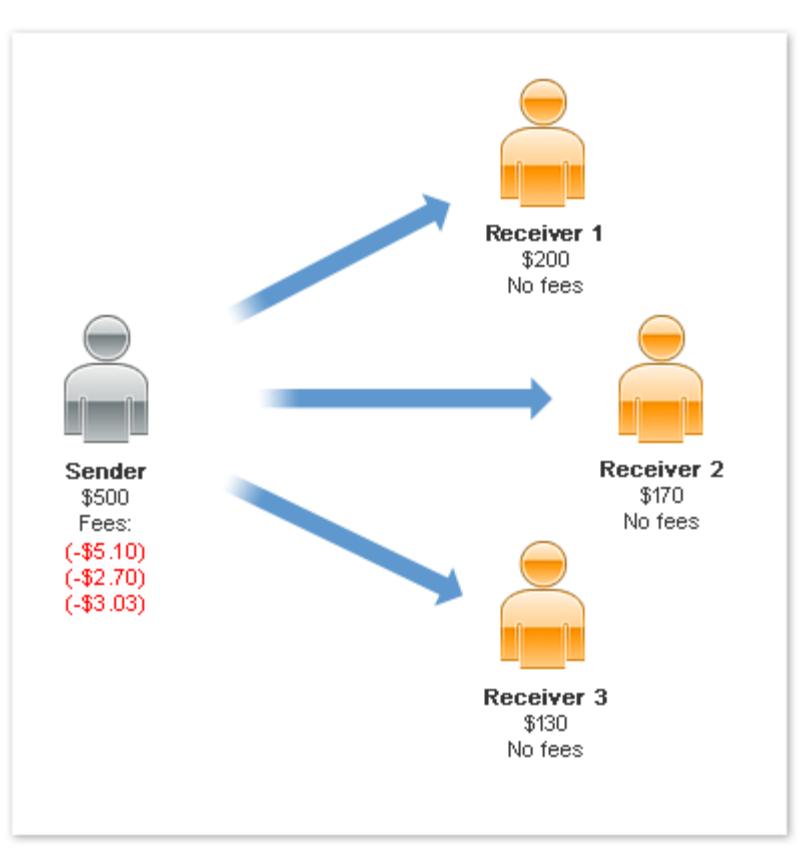
You can specify who pays these fees. Fee payment configurations include:

- Sender Pays the Fee
- Receiver Pays the Fee in a Parallel Payment
- Each Receiver Pays the Fee in a Chained Payment
- Primary Receiver Pays the Fee in a Chained Payment
- Secondary Receivers Pay the Fee in a Chained Payment

NOTE: Fees are determined by PayPal and are based on criteria, such as the transaction volume of the receiver. In the examples that follow, the fees shown are representative only and not actual fees.

Sender Pays the Fee

The sender can pay a fee for a simple payment, parallel payment, or a chained payment. The following example shows fees being paid by the sender of a parallel payment, based on the assessment for each receiver:



In this example, the sender pays a total of \$510.83, which includes all fees.

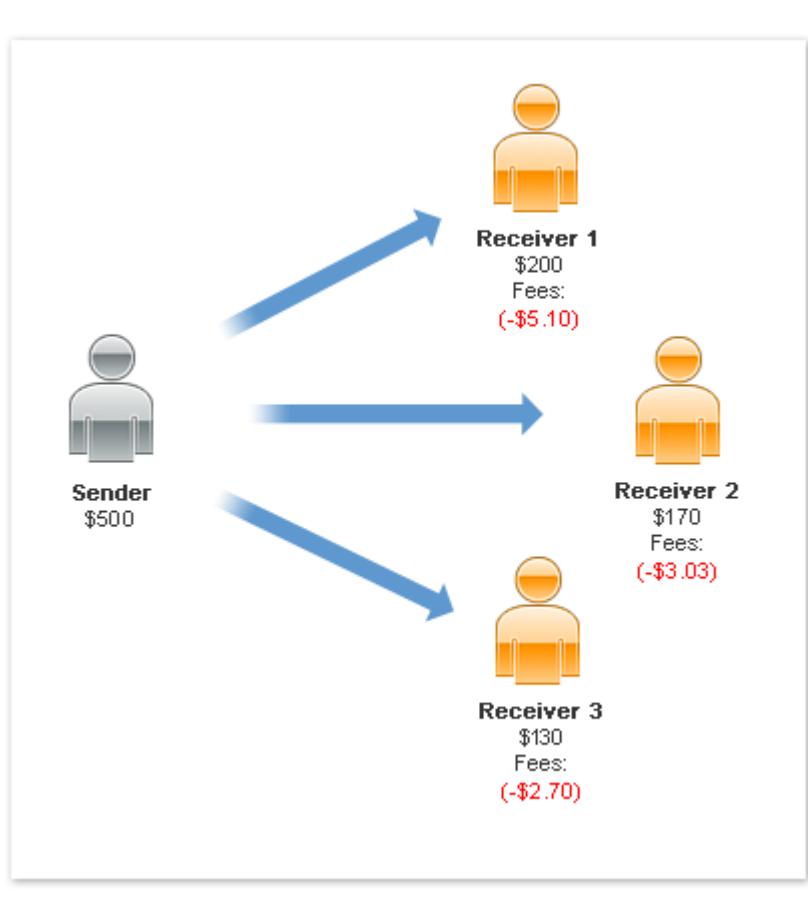
Introducing Adaptive Payments

Fee Payment Configuration

NOTE: The scenario above is an example only and is not representative of actual PayPal fees.

Receiver Pays the Fee in a Parallel Payment

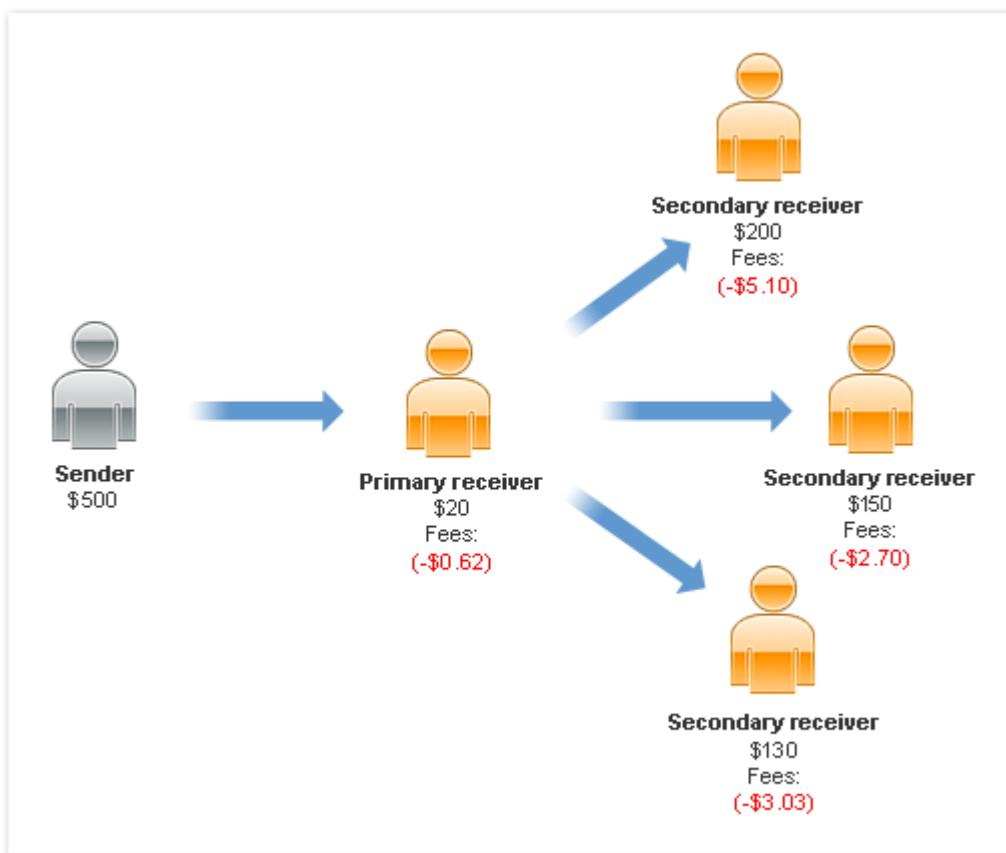
If the receivers pay the fee in a parallel payment, each receiver pays a portion of the fee, based on their assessment. The following example shows the receivers paying the fees:



NOTE: The scenario above is an example only and is not representative of actual PayPal fees.

Each Receiver Pays the Fee in a Chained Payment

If the receivers pay the fee in a chained payment, each receiver pays a portion of the fee, based on their assessment. The following example shows the receivers paying the fees:



In this example, the primary receiver, identified as the merchant, pays a fee for \$20 received. Each of the other receivers also pay a fee on the amount each receives.

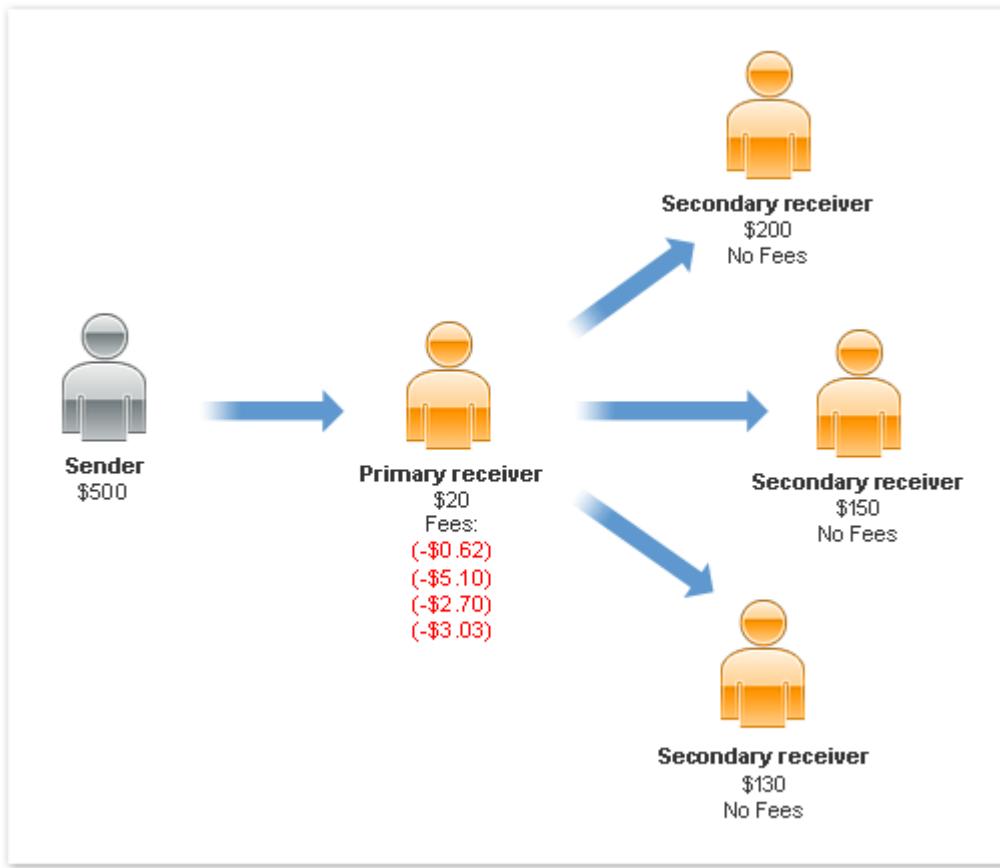
NOTE: The scenario above is an example only and is not representative of actual PayPal fees.

Primary Receiver Pays the Fee in a Chained Payment

If only the primary receiver pays the fee in a chained payment, other receivers pay no fees. The fees paid by the primary receiver, however, are based upon the total fees assigned to all receivers. The following example shows only the primary receiver, identified as the merchant, paying all fees:

Introducing Adaptive Payments

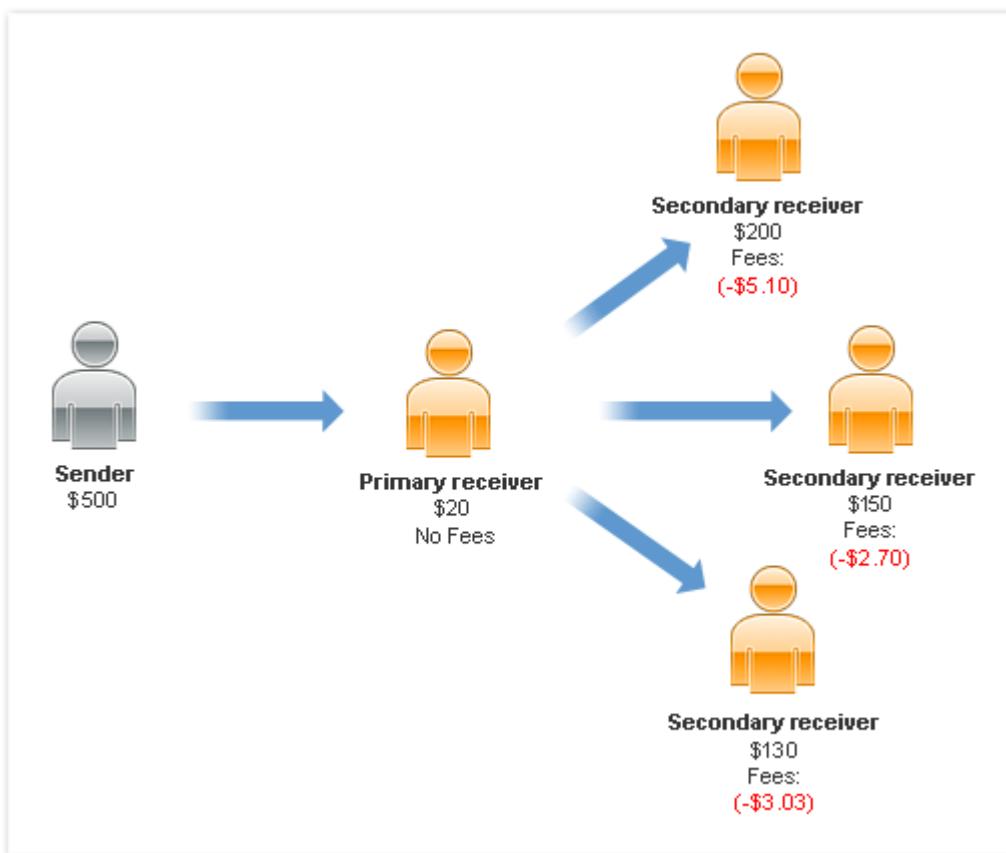
Fee Payment Configuration



NOTE: The scenario above is an example only and is not representative of actual PayPal fees.

Secondary Receivers Pay the Fee in a Chained Payment

For chained payments, the primary receiver is not obliged to pay the fees. It is possible for the secondary receivers to pay all fees as shown in the following diagram:



NOTE: The scenario above is an example only and is not representative of actual PayPal fees.

Introducing Adaptive Payments

Fee Payment Configuration

2

Getting Started

These basic scenarios get you up and running quickly with the Adaptive Payments API. The sample code shows different combinations of requests with different bindings in various programming languages.

Adaptive Payments API Operations

Adaptive Payments provides API operations, enabling you to build an application that handles payments, preapprovals for payments, refunds, and additional operations related to payments. Some kinds of payments and operations require specific permission to use.

API Operations Related to Payments

API Operation	Description
Pay	Transfers funds from a sender's PayPal account to one or more receivers' PayPal accounts (up to 6 receivers)
PaymentDetails	Obtains information about a payment set up with the Pay API operation
ExecutePayment	Executes a payment
GetPaymentOptions	Obtain the settings specified with the SetPaymentOptions API operation
SetPaymentOptions	Sets payment options

API Operations Related to Preapprovals

API Operation	Description
Preapproval	Sets up preapprovals, which is an approval to make future payments on the sender's behalf
PreapprovalDetails	Obtains information about a preapproval
CancelPreapproval	Cancels a preapproval
ConfirmPreapproval	Confirms that the specified preapproval can be used to make payments

Other API Operations

API Operation	Description
Refund	Refunds all or part of a payment
ConvertCurrency	Obtains the current foreign exchange (FX) rate for a specific amount and currency
GetFundingPlans	Determines the funding sources that are available for a specified payment
GetShippingAddresses	Obtains the selected shipping address
GetAvailableShippingAddresses	Obtains available shipping addresses

Adaptive Payments Endpoints

The endpoint is determined by the API operation and the environment in which you want to execute the API operation. For example, if you want to send a Pay request to the sandbox endpoint, specify the following URL:

`https://svcs.sandbox.paypal.com/AdaptivePayments/Pay`

You can specify the following endpoints:

Environment	Endpoint
Production	<code>https://svcs.paypal.com/AdaptivePayments/API_operation</code>
Sandbox	<code>https://svcs.sandbox.paypal.com/AdaptivePayments/API_operation</code>
Beta sandbox	<code>https://svcs.beta-sandbox.paypal.com/AdaptivePayments/API_operation</code>

HTTP Headers

Each request message includes HTTP headers specifying authentication, the application ID, the device ID or IP address, and the payload format or protocol (SOAP).

Adaptive Payments supports request bodies with JSON, NVP, and XML data formats for REST implementations. You can specify different formats for the request and response, such as sending the request in JSON and requesting an XML response.

For SOAP, you must also include a specific SOAP protocol header (see the SOAP messages section).

The following is an example of HTTP headers for NVP in Java for a web implementation:

```
headers.put("X-PAYPAL-SECURITY-USERID", "tok261_biz_api.abc.com");
headers.put("X-PAYPAL-SECURITY-PASSWORD", "1244612379");
headers.put("X-PAYPAL-SECURITY-SIGNATURE", "lkfg9groingghb4uw5"
headers.put("X-PAYPAL-DEVICE-IPADDRESS", "168.212.226.204");
headers.put("X-PAYPAL-REQUEST-DATA-FORMAT", "NV");
headers.put("X-PAYPAL-RESPONSE-DATA-FORMAT", "NV");
headers.put("X-PAYPAL-APPLICATION-ID", "APP-80W284485P519543T");
```

NOTE: HTTP headers are case sensitive.

Authentication

Use your PayPal account API credentials to authenticate your application. Your API credentials include an API username and API password. If you are using 3-token authentication, you must also specify an API signature. If you are using a certificate, the certificate is used with the username and password; the signature is not used. To specify API credentials, include the following HTTP headers in your request message (observing case sensitivity):

HTTP Headers for Authentication

Header	Description
X-PAYPAL-SECURITY-USERID	Your API username
X-PAYPAL-SECURITY-PASSWORD	Your API password
X-PAYPAL-SECURITY-SIGNATURE	Your API signature, which is required only if you use 3-token authorization; a certificate does not use a signature
X-PAYPAL-SECURITY-SUBJECT	Third-party permission specification, which specifies the email address or phone number (for mobile) of the party on whose behalf you are calling the API operation. The subject must grant you third-party access in their PayPal profile. NOTE: Resources specified by the API operation, such as a payment or preapproval identified by a key, must be owned by the subject granting the third-party permission.

Specifying JSON, NVP, or XML Data Formats

Use the HTTP header X-PAYPAL-REQUEST-DATA-FORMAT to specify the data format the request body. You can send messages using JSON, NVP or straight XML.

Use the and X-PAYPAL-RESPONSE-DATA-FORMAT headers to specify the data format for the response.

For SOAP messages, refer to the next section.

HTTP Headers for JSON, NVP, and XML Data Formats

Header	Description
X-PAYPAL-REQUEST-DATA-FORMAT	The payload format for the request. Allowable values are: <ul style="list-style-type: none">• NV – Name-value pairs• XML – Extensible markup language• JSON – JavaScript object notation
X-PAYPAL-RESPONSE-DATA-FORMAT	The payload format for the response. Allowable values are: <ul style="list-style-type: none">• NV – Name-value pairs• XML – Extensible markup language• JSON – JavaScript object notation

SOAP Messages

To use Adaptive Payments with SOAP, include the HTTP headers for authentication as described in the section Authentication and the application ID as described in the next section.

In addition, include the X-PAYPAL-MESSAGE-PROTOCOL header with a SOAP11 value.

The following is a header example for an Adaptive Payments API call for a SOAP message:

```
headers.put("X-PAYPAL-SECURITY-USERID", "tok261_biz_api.abc.com");
headers.put("X-PAYPAL-SECURITY-PASSWORD", "1244612379");
headers.put("X-PAYPAL-SECURITY-SIGNATURE", "lkfg9groingghb4uw5");
headers.put("X-PAYPAL-DEVICE-IPADDRESS", "168.212.226.204");
headers.put("X-PAYPAL-MESSAGE-PROTOCOL", "SOAP11");
headers.put("X-PAYPAL-APPLICATION-ID", "APP-80W284485P519543T");
```

Below are the service name, port type, binding and location for SOAP as defined in the Adaptive Payments WSDL.

```
<wsdl:service name="AdaptivePayments">
  <wsdl:port name="AdaptivePaymentsSOAP11_http"
    <binding="services:AdaptivePaymentsSOAP11Binding">
      <soap:address location="https://svcs.paypal.com/AdaptivePayments" />
```

Specifying Application and Device Information

You also must identify the application. You can optionally identify other information associated with the client and the API version:

HTTP Headers for Application and Device identification

Header	Description
X-PAYPAL-APPLICATION-ID	(Required) Your application's identification, which is issued by PayPal. NOTE: Check X.com for which application ID must be defined for working in the sandbox.
X-PAYPAL-DEVICE-ID	(Optional) Client's device ID, such as a mobile device's IMEI number or a web browser cookie.
X-PAYPAL-DEVICE-IPADDRESS	(Required) Client's IP address.
X-PAYPAL-SERVICE-VERSION	(Optional) The version of an API operation to use. By default, PayPal executes a request with the current version of an API operation. NOTE: PayPal recommends not specifying a version unless it is absolutely required.

Making a Simple Payment (JSON)

A *simple* payment is when a sender (whose account is debited) sends a payment (amount and currency) to a single receiver (whose account is credited).

In the example below, Julie pays David \$10.00. The following event sequence takes place:

- You send a PayRequest.
- You receive a response with a pay key.
- You use the _ap-payment command to redirect Julie to PayPal to log in and approve the payment.

Pay Request for Simple Payment

```
{
  "returnUrl": "http://example.com/returnURL.htm", \
  "requestEnvelope": {
    "errorLanguage": "en_US",
    "currencyCode": "USD",
    "receiverList": {
      "receiver": [
        {
          "email": "david@example.com",
          "amount": "10.00"
        }
      ]
    },
    "cancelUrl": "http://example.com/cancelURL.htm",
    "actionType": "PAY"
}
```

Pay Response for Simple Payment

```
{
  "responseEnvelope": {
    "timestamp": "2009-10-06T14:30:39.383-07:00",
    "ack": "Success",
    "correlationId": "cfe8f8783f1d3",
    "build": "DEV"
  },
  "payKey": "AP-17266198048308436",
  "paymentExecStatus": "CREATED"
}
```

Getting Started

Making a Parallel Payment (NVP)

The response includes a **pay key**, which is a token you use in subsequent calls to Adaptive Payments APIs to identify this particular payment.

In this particular scenario, the `paymentExecStatus` variable is set to `CREATED` instead of `COMPLETED`, which indicates that the payment has been created, but has not yet been executed.

Redirecting Sender for Payment Approval

Use the `_ap-payment` command redirect Julie to PayPal to log in and approve the payment:

`https://www.paypal.com/webscr?cmd=_ap-payment&paykey=AP-17266198048308436`

Making a Parallel Payment (NVP)

A *parallel* payment is when a sender (whose account is debited) sends a single payment (amount and currency) up to 6 receivers. The sender can see each payment to a receiver.

In the example below, Paul makes a single payment of \$14, which is split into a \$9 payment to Andrea and a \$5 payment to Linda. The following event sequence takes place:

- You send a `PayRequest`, specifying an amount to be paid for each receiver.
- You receive a response with a pay key.
- You use the `_ap-payment` command to redirect Paul to PayPal to log in and approve the payment.

Pay Request for Parallel Payment

```
&actionType=PAY
&cancelUrl=http:\\example.com\cancel.htm
&currencyCode=USD
&receiverList.receiver(0).amount=9.00
&receiverList.receiver(0).email=andrea@example.com
&receiverList.receiver(1).amount=5.00
&receiverList.receiver(1).email=linda@example.com
&requestEnvelope.errorLanguage=en_US
&returnUrl=http:\\example.com\return.htm
```

Pay Response for Parallel Payment

```
responseEnvelope.timestamp=2009-11-03T08%3A12.937-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=b1cc3eabfa4c1
&responseEnvelope.build=942345
&payKey=AP-688241038Y786593D
&paymentExecStatus=CREATED
```

The response includes a **pay key**, which is a token you use in subsequent calls to Adaptive Payments APIs to identify this particular payment.

In this particular scenario, the `paymentExecStatus` variable is set to `CREATED` instead of `COMPLETED`, which indicates that the payment has been created, but has not yet been executed.

Redirecting Sender for Payment Approval

Use the `_ap-payment` command redirect Paul to PayPal to log in and approve the payment:

```
https://www.paypal.com/webscr?cmd=_ap-payment&paykey=AP-688241038Y786593D
```

Making a Chained Payment (XML)

A *chained* payment is when a sender sends a payment to a PayPal-registered receiver who is the primary receiver.

In the example below, Tim makes a single payment of \$100 to Frank, who is the primary receiver. Of this amount, Frank keeps \$25 and pays another Yvonne \$75. The following event sequence takes place:

- You send a `PayRequest`, enabling the primary receiver.
- You receive a response with a pay key.
- You use the `_ap-payment` command to redirect Tim to PayPal to log in and approve the payment.

With chained payments, the sender only sees the transaction to the primary API caller. The receiver only sees the transaction from the primary API caller. The transactions from and to the receivers are hidden from the sender and receivers.

Getting Started

Making a Chained Payment (XML)

Pay Request for Chained Payment

```
<PayRequest>
  <requestEnvelope>
    <errorLanguage>en_US</errorLanguage> </requestEnvelope>
  <cancelUrl xmlns="">http://example.com/cancelURL.htm</cancelUrl>
  <actionType>PAY</actionType>
  <currencyCode xmlns="">USD</currencyCode>
  <receiverList xmlns="">
    <receiver>
      <amount>100</amount>
      <email>frank@example.com</email>
      <primary>true</primary>
    </receiver>
    <receiver>
      <amount>75</amount>
      <email>yvonne@example.com</email>
      <primary>false</primary>
    </receiver>
  </receiverList>
  <returnUrl xmlns="">http://example.com/returnURL.htm</returnUrl>
</PayRequest>
```

Pay Response for Chained Payment

```
<?xml version='1.0' encoding='UTF-8'?>
<ns2:PayResponse xmlns:ns2="http://svcs.paypal.com/types/ap">
  <responseEnvelope>
    <timestamp>2009-10-06T17:24:03.874-07:00</timestamp>
    <ack>Success</ack><correlationId>eca3a204200f4</correlationId>
    <build>1044393</build></responseEnvelope>
    <payKey>AP-688241038Y786593D</payKey>
    <paymentExecStatus>CREATED</paymentExecStatus></ns2:PayResponse>
```

The response includes a **pay key**, which is a token you use in subsequent calls to Adaptive Payments APIs to identify this particular payment.

In this particular scenario, the `paymentExecStatus` variable is set to `CREATED` instead of `COMPLETED`, which indicates that the payment has been created, but has not yet been executed.

Redirecting Sender for Payment Approval

Use the `_ap-payment` command redirect Tim to PayPal to log in and approve the payment:

```
https://www.paypal.com/webscr?cmd=_ap-payment&paykey=AP-688241038Y786593D
```

3

Pay API Operation

Use the Pay API operation to transfer funds from a sender's PayPal account to one or more receivers' PayPal accounts. You can use the Pay API operation to make simple payments, chained payments, or parallel payments; these payments can be explicitly approved, preapproved, or implicitly approved.

Pay Summary

Create your PayRequest message by setting the common fields. If you want more than a simple payment, add fields for the specific kind of request, which include parallel payments, chained payments, implicit payments, and preapproved payments.

Common Fields for All Payments

For each kind of payment, you must specify values for the following fields:

Field	Description
actionType	The action for this request. Possible values are: <ul style="list-style-type: none">• PAY – Use this option if you are not using the Pay request in combination with ExecutePayment.• CREATE – Use this option to set up the payment instructions with SetPaymentOptions and then execute the payment at a later time with the ExecutePayment.• PAY_PRIMARY – For chained payments only, specify this value to delay payments to the secondary receivers; only the payment to the primary receiver is processed.
receiverList.receiver(0).email	A receiver's email address
receiverList.receiver(0).amount	Amount to be credited to the receiver's account
currencyCode	The code for the currency in which the payment is made; you can specify only one currency, regardless of the number of receivers
cancelUrl	URL to redirect the sender's browser to after canceling the approval for a payment; it is always required but only used for payments that require approval (explicit payments)

Pay API Operation

Pay Summary

Field	Description
returnUrl	URL to redirect the sender's browser to after the sender has logged into PayPal and approved a payment; it is always required but only used if a payment requires explicit approval
requestEnvelope.errorLanguage	The code for the language in which errors are returned, which must be en_US.

Parallel Payments

For a simple payment, you must specify values for the fields above. For a parallel payment, you must specify both the above values and the following fields:

Field	Description
receiverList.receiver(<i>n</i>).email	The receiver's email address for each receiver, where <i>n</i> is between 0 and 5 for a maximum of 6 receivers
receiverList.receiver(<i>n</i>).amount	Amount to send to each corresponding receiver

Chained Payments

For a chained payment, you must specify all required fields for a parallel payment and you must also specify enable the primary receiver:

Field	Description
receiverList.receiver(<i>n</i>).email	The receivers' email addresses, where <i>n</i> is between 0 and 5 for a total of one primary receiver and between one and 5 secondary receivers
receiverList.receiver(<i>n</i>).amount	Amount to send to each corresponding receiver
receiverList.receiver(<i>n</i>).primary	Set to true to indicate a chained payment; only one receiver can be a primary receiver. Omit this field, or set it to false for simple and parallel payments.

By default, a payment requires explicit approval in which the sender must log in to PayPal. In that case, you are not required to specify the sender's email address. To initiate the approval process, you must redirect the sender to PayPal as follows:

`https://www.paypal.com/webscr?cmd=_ap-payment&paykey=`*value*

NOTE: The command is _ap-payment. You obtain the value of the paykey parameter from the payKey field in the Pay API operation's response message.

Implicit Payments

If you are the API caller and you specify your email address in the `senderEmail` field, PayPal implicitly approves the payment without redirecting to PayPal:

Field	Description
<code>senderEmail</code>	Sender's email address

Preapproved Payments

If the sender has set up a preapproval, you can use the preapproval to avoid explicit approval. In that case, you must specify values for the fields below.

Field	Description
<code>preapprovalKey</code>	Preapproval key for the approval set up between you and the sender
<code>pin</code>	Sender's personal identification number, if one was specified when the sender agreed to the approval

Payments for Digital Goods

You handle payments for digital goods in the same way you handle payments for other goods and services, with the following exceptions:

- To specify a payment for digital goods, you must specify `DIGITALGOODS` for each receiver in your receiver list; specify `receiverList.receiver(n).paymentType=DIGITALGOODS` for each receiver, where *n* identifies the receiver, starting with 0.
- If you specify a payment for digital goods, you cannot specify a `senderEmail` address or include a funding constraint.
- You must redirect the sender to the following PayPal URL to complete the payment for digital goods:
`https://paypal.com/webapps/adaptivepayment/flow/pay?paykey=....`

Payment Notifications

Notifications are sent after payment is executed, which follows approval of the payment by the sender if required:

- PayPal sends an email to the sender and all receivers associated with a payment when the transfer is complete; you receive an email only if you are also the payment sender or receiver.

- When the payment is complete, PayPal sends an IPN message to the URL specified in the `ipnNotificationUrl` field of the Pay request.

Additional Notes About the Pay API Operation

- With parallel payments, fund transfers to some receivers may occur before others. Therefore, set `reverseAllParallelPaymentsOnError` to `true`. In the event of an error, this setting guarantees that transfers to all receivers are reversed and all funds are returned to the sender. If `reverseAllParallelPaymentsOnError` is disabled, completed transfers are not reversed and funds that have already been transferred are no longer available to the sender.
- You can specify your own unique tracking ID in the `trackingID` field and use this value to obtain information about a payment or to request a refund. The tracking ID is provided as a convenience in cases when you already maintain an ID that you want to associate with a payment. You can also track payments using the `paykey`.
- You can use the Pay API operation to make *unilateral* payments under limited circumstances. A unilateral payment is a payment that is made to a receiver who does not have a PayPal account. Unilateral payments can be used with simple or parallel payments that are implicit or preapproved. Unilateral payments are not designed to be used with chained payments or payments that require manual approval through the web flow.

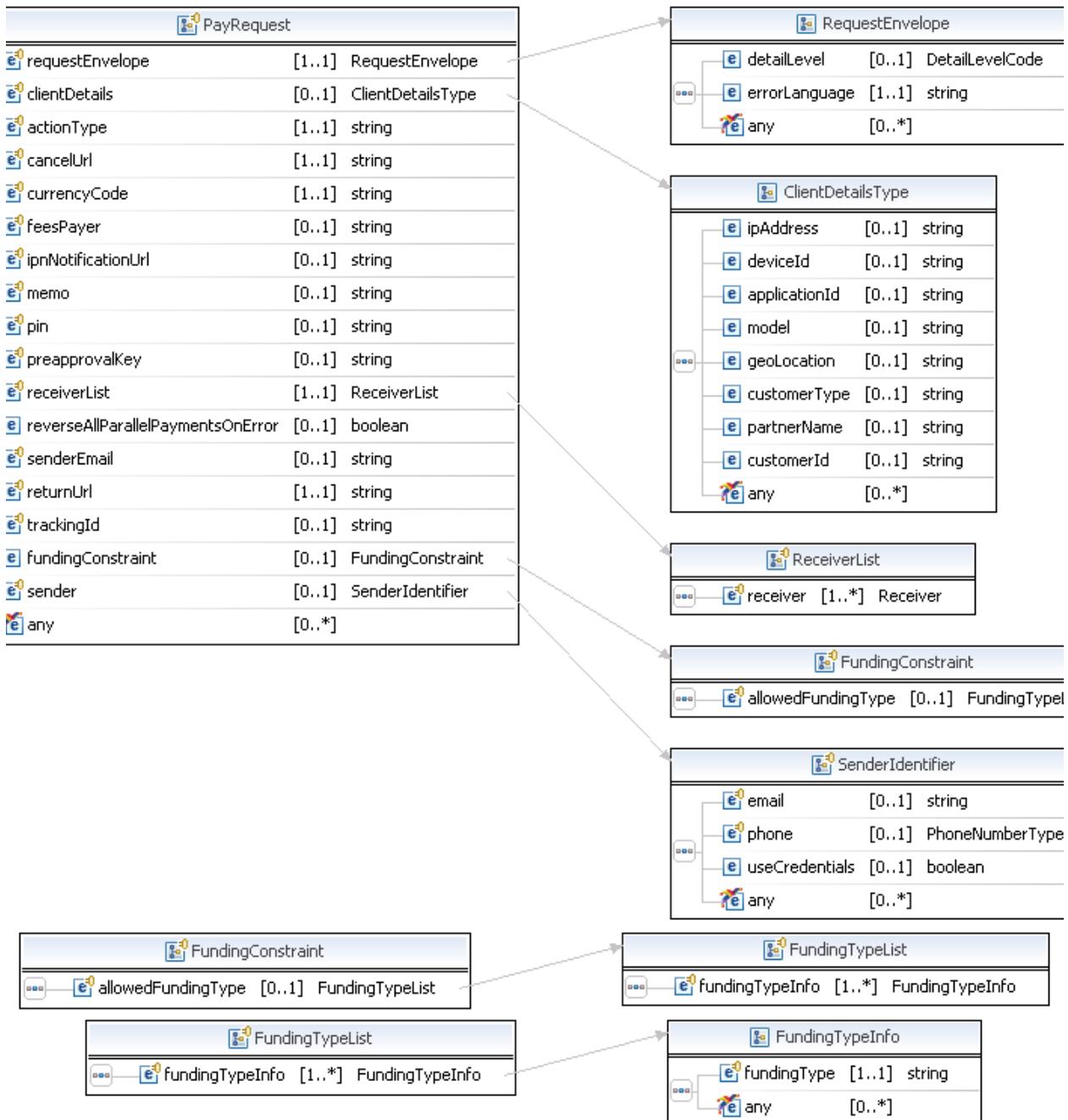
When you send a unilateral payment, you send a payment request that includes an email address for a receiver, and this email address is not linked to a registered PayPal account. The receiver receives an email notifying the receiver to create an account and claim the payment.

PayPal holds a payment to a receiver whose email address is not yet registered or confirmed until the receiver creates a PayPal account and confirms the email address. If a refund specifies a receiver whose email address is not yet registered or confirmed, the payment to the receiver is canceled.

- For a guest payment, do not set the sender's email address or the preapproval key. Specifying the sender's email address prevents the sender from choosing the guest payment option. Implicit approval and preapproval options are not allowed. Each receiver of a guest payment must be a verified PayPal business or premier account holder.
- You can specify a value for either `senderEmail` or one of the `SenderId` fields. If you use `SenderId` to identify the sender, you can only specify `SenderId.email` or set `SenderId.useCredentials` to `true`, but not both; `SenderId.phone` is reserved for future use.
- Specify the `senderEmail` or `SenderId.email` to set up a payment where the sender and the API caller are the same (implicit approval) or for a preapproved payment.
- If the `senderEmail` or `SenderId.email` is specified in the request, the email address must be registered with [paypal.com](#). If you do not include one of these fields in the payment request, the sender can either log in using an existing account (that is not associated with a receiver) or create a new account during the payment flow.

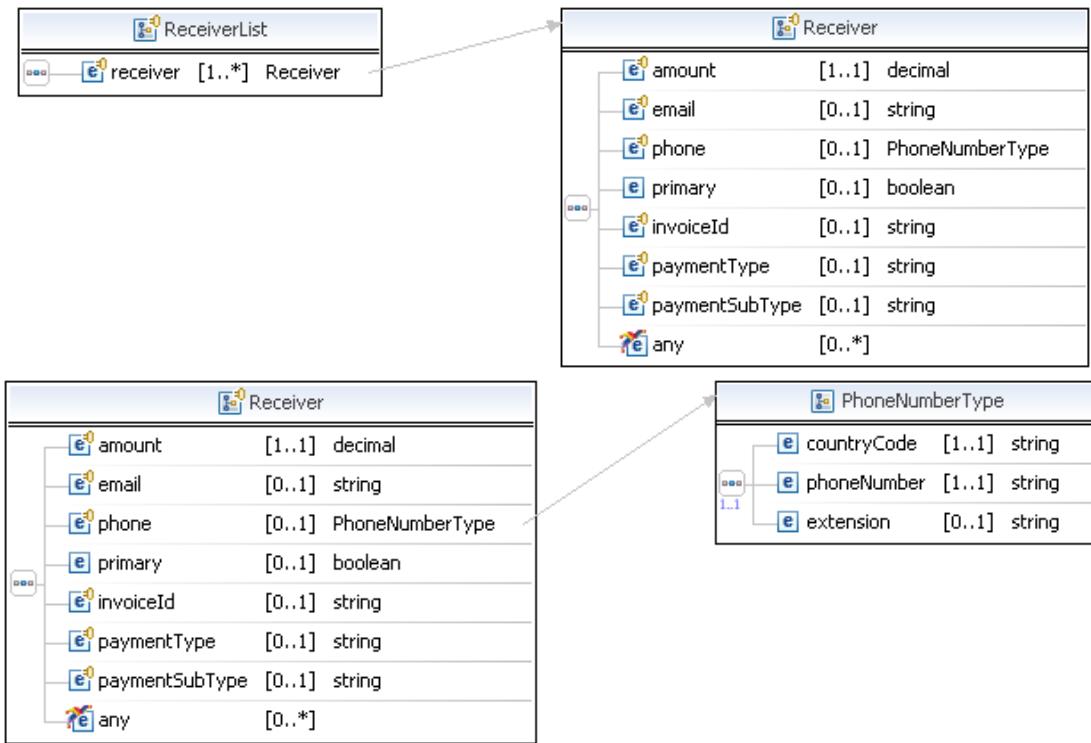
PayRequest Message

The PayRequest message contains the instructions required to make a payment.



Pay API Operation

PayRequest Message



PayRequest Fields

Field	Description
<code>actionType</code>	<code>xs:string</code> <i>(Required)</i> Whether the Pay request pays the receiver or whether the Pay request is set up to create a payment request, but not fulfill the payment until the ExecutePayment is called. Allowable values are: <ul style="list-style-type: none">PAY – Use this option if you are not using the Pay request in combination with ExecutePayment.CREATE – Use this option to set up the payment instructions with SetPaymentOptions and then execute the payment at a later time with the ExecutePayment.PAY_PRIMARY – For chained payments only, specify this value to delay payments to the secondary receivers; only the payment to the primary receiver is processed.
<code>cancelUrl</code>	<code>xs:string</code> <i>(Required)</i> The URL to which the sender's browser is redirected if the sender cancels the approval for the payment after logging in to paypal.com to approve the payment. Specify the URL with the HTTP or HTTPS. Maximum length: 1024 characters

Field	Description
clientDetails	common:ClientDetailsType <i>(Optional)</i> Information about the sender.
currencyCode	xs:string <i>(Required)</i> The currency code. Allowable values are: <ul style="list-style-type: none">• Australian Dollar – AUD• Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none">• Canadian Dollar – CAD• Czech Koruna – CZK• Danish Krone – DKK• Euro – EUR• Hong Kong Dollar – HKD• Hungarian Forint – HUF• Israeli New Sheqel – ILS• Japanese Yen – JPY• Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none">• Mexican Peso – MXN• Norwegian Krone – NOK• New Zealand Dollar – NZD• Philippine Peso – PHP• Polish Zloty – PLN• Pound Sterling – GBP• Singapore Dollar – SGD• Swedish Krona – SEK• Swiss Franc – CHF• Taiwan New Dollar – TWD• Thai Baht – THB• U.S. Dollar – USD
feesPayer	xs:string <i>(Optional)</i> The payer of PayPal fees. Allowable values are: <ul style="list-style-type: none">• SENDER – Sender pays all fees (for personal, implicit simple/parallel payments; do not use for chained or unilateral payments)• PRIMARYRECEIVER – Primary receiver pays all fees (chained payments only)• EACHRECEIVER – Each receiver pays their own fee (default, personal and unilateral payments)• SECONDARYONLY – Secondary receivers pay all fees (use only for chained payments with one secondary receiver)

Pay API Operation

PayRequest Message

Field	Description
fundingConstraint	<p>ap:FundingConstraint</p> <p>(Optional) Specifies a list of allowed funding types for the payment. This is a list of funding selections that can be combined in any order to allow payments to use the indicated funding type. If this Parameter is omitted, the payment can be funded by any funding type that is supported for Adaptive Payments.</p> <p>NOTE: FundingConstraint is unavailable to API callers with standard permission levels; for more information, refer to the section Adaptive Payments Permission Levels.</p>
ipnNotificationUrl	<p>xs:string</p> <p>(Optional) The URL to which you want all IPN messages for this payment to be sent.</p> <p>Maximum length: 1024 characters</p>
memo	<p>xs:string</p> <p>(Optional) A note associated with the payment (text, not HTML).</p> <p>Maximum length: 1000 characters, including newline characters</p>
pin	<p>xs:string</p> <p>(Optional) The sender's personal identification number, which was specified when the sender signed up for a preapproval.</p>
preapprovalKey	<p>xs:string</p> <p>(Optional) The key associated with a preapproval for this payment. The preapproval key is required if this is a preapproved payment.</p> <p>NOTE: The Preapproval API is unavailable to API callers with Standard permission levels.</p>
receiverList	<p>ap:ReceiverList</p> <p>(Required) Information about the receivers of the payment.</p>
requestEnvelope	<p>common:RequestEnvelope</p> <p>(Required) Information common to each Method, such as the language in which an error message is returned.</p>
returnUrl	<p>xs:string</p> <p>(Required) The URL to which the sender's browser is redirected after approving a payment on paypal.com. Specify the URL with the HTTP or HTTPS designator.</p> <p>Maximum length: 1024 characters</p>
reverseAllParallelPaymentsOnError	<p>xs:boolean</p> <p>(Optional) Whether to reverse parallel payments if an error occurs with a payment.</p> <p>Allowable values are:</p> <ul style="list-style-type: none">• true – Each parallel payment is reversed if an error occurs• false – Only incomplete payments are reversed (default)

Field	Description
senderEmail	xs:string <i>(Optional)</i> Sender's email address. Maximum length: 127 characters
sender	ap:SenderIdentifier <i>(Optional)</i> Sender's identifying information.
trackingId	xs:string <i>(Optional)</i> A unique ID that you specify to track the payment. NOTE: You are responsible for ensuring that the ID is unique. Maximum length: 127 characters

ClientDetails Fields

Field	Description
applicationId	xs:string <i>(Optional)</i> Your application's identification, such as the name of your application
customerId	xs:string <i>(Optional)</i> Your ID for this sender Maximum length: 127 characters
customerType	xs:string <i>(Optional)</i> Your identification of the type of customer Maximum length: 127 characters
deviceId	xs:string <i>(Optional)</i> Sender's device ID, such as a mobile device's IMEI number or a web browser cookie. If a device ID was passed with the PayRequest, use the same ID here. Maximum length: 127 characters
geoLocation	xs:string <i>(Optional)</i> Sender's geographic location Maximum length: 127 characters
ipAddress	xs:string <i>(Optional)</i> Sender's IP address. If an IP address was passed with the PayRequest, use the same ID here.
model	xs:string <i>(Optional)</i> A sub-identification of the application Maximum length: 127 characters

Pay API Operation

PayRequest Message

Field	Description
partnerName	<code>xs:string</code> <i>(Optional)</i> Your organization's name or ID Maximum length: 127 characters

FundingConstraint Fields

Field	Description
allowedFundingType	<code>ap:FundingTypeList</code> <i>(Optional)</i> Specifies a list of allowed funding selections for the payment. This is a list of funding selections that can be combined in any order to allow payments to use the indicated funding type. If this field is omitted, the payment can be funded by any funding type that is supported for Adaptive Payments. NOTE: FundingConstraint is unavailable to API callers with standard permission levels; for more information, refer to the section Adaptive Payments Permission Levels. NOTE: To use iACH, omit this field and do not specify a funding source for the payment.

FundingTypeList Fields

Field	Description
fundingTypeInfo	<code>ap:FundingTypeInfo</code> <i>(Optional)</i> Specifies a list of allowed funding selections for the payment. This is a list of funding selections that can be combined in any order to allow payments to use the indicated funding type. If this field is omitted, the payment can be funded by any funding type that is supported for Adaptive Payments. NOTE: FundingConstraint is unavailable to API callers with standard permission levels; for more information, refer to the section Adaptive Payments Permission Levels.

FundingTypeInfo Fields

Field	Description
fundingType	<p>xs:string</p> <p>(Required) Specifies a list of allowed funding selections for the payment. This is a list of funding selections that can be combined in any order to allow payments to use the indicated funding type. If this field is omitted, the payment can be funded by any funding type that is supported for Adaptive Payments.</p> <p>Allowable values are:</p> <ul style="list-style-type: none"> • ECHECK – Electronic check • BALANCE – PayPal account balance • CREDITCARD – Credit card <p>NOTE: ECHECK and CREDITCARD include BALANCE implicitly.</p> <p>NOTE: FundingConstraint is unavailable to API callers with standard permission levels; for more information, refer to the section Adaptive Payments Permission Levels.</p>

ReceiverList Fields

Field	Description
receiver	<p>ap:Receiver</p> <p>(Required) Receiver is the party whose account is credited.</p>

Receiver Fields

Field	Description
amount	<p>xs:decimal</p> <p>(Required) Amount to be paid to the receiver.</p>
email	<p>xs:string</p> <p>Receiver's email address. This address can be unregistered with paypal.com. If so, a receiver cannot claim the payment until a PayPal account is linked to the email address. The PayRequest must pass either an email address or a phone number.</p> <p>Maximum length: 127 characters</p>
invoiceId	<p>xs:string</p> <p>(Optional) The invoice number for the payment. This data in this field shows on the Transaction Details report.</p> <p>Maximum length: 127 characters</p>

Pay API Operation

PayRequest Message

Field	Description
paymentType	<p>xs:string <i>(Optional)</i> The transaction type for the payment.</p> <p>Allowable values are:</p> <ul style="list-style-type: none">• GOODS – This is a payment for non-digital goods• SERVICE – This is a payment for services (default)• PERSONAL – This is a person-to-person payment• CASHADVANCE – This is a person-to-person payment for a cash advance• DIGITALGOODS – This is a payment for digital goods <p>NOTE: Person-to-person payments are valid only for parallel payments that have the feesPayer field set to EACHRECEIVER or SENDER.</p>
paymentSubType	<p>xs:string <i>(Optional)</i> The transaction subtype for the payment.</p>
phone	<p>common:PhoneNumberType</p> <p>A type to specify the receiver's phone number. The PayRequest must pass either an email address or a phone number as the payment receiver.</p>
primary	<p>xs:boolean <i>(Optional)</i> Whether this receiver is the primary receiver, which makes the payment a chained payment. You can specify at most one primary receiver. Omit this field for simple and parallel payments.</p> <p>Allowable values are:</p> <ul style="list-style-type: none">• true – Primary receiver• false – Secondary receiver (default)

PhoneNumberType Fields

Field	Description
countryCode	<p>xs:string <i>(Required)</i> Telephone country code.</p>
phoneNumber	<p>xs:string <i>(Required)</i> Telephone number.</p>
extension	<p>xs:string <i>(Optional)</i> Telephone extension.</p>

SenderIdentifier Fields

Field	Description
email	<p>xs:string <i>(Optional)</i> Sender's email address. Maximum length: 127 characters</p>
phone	<p>common:PhoneNumberType <i>(Optional)</i> Sender's phone number.</p>
useCredentials	<p>xs:boolean <i>(Optional)</i> If true, use credentials to identify the sender; default is false.</p>

RequestEnvelope Fields

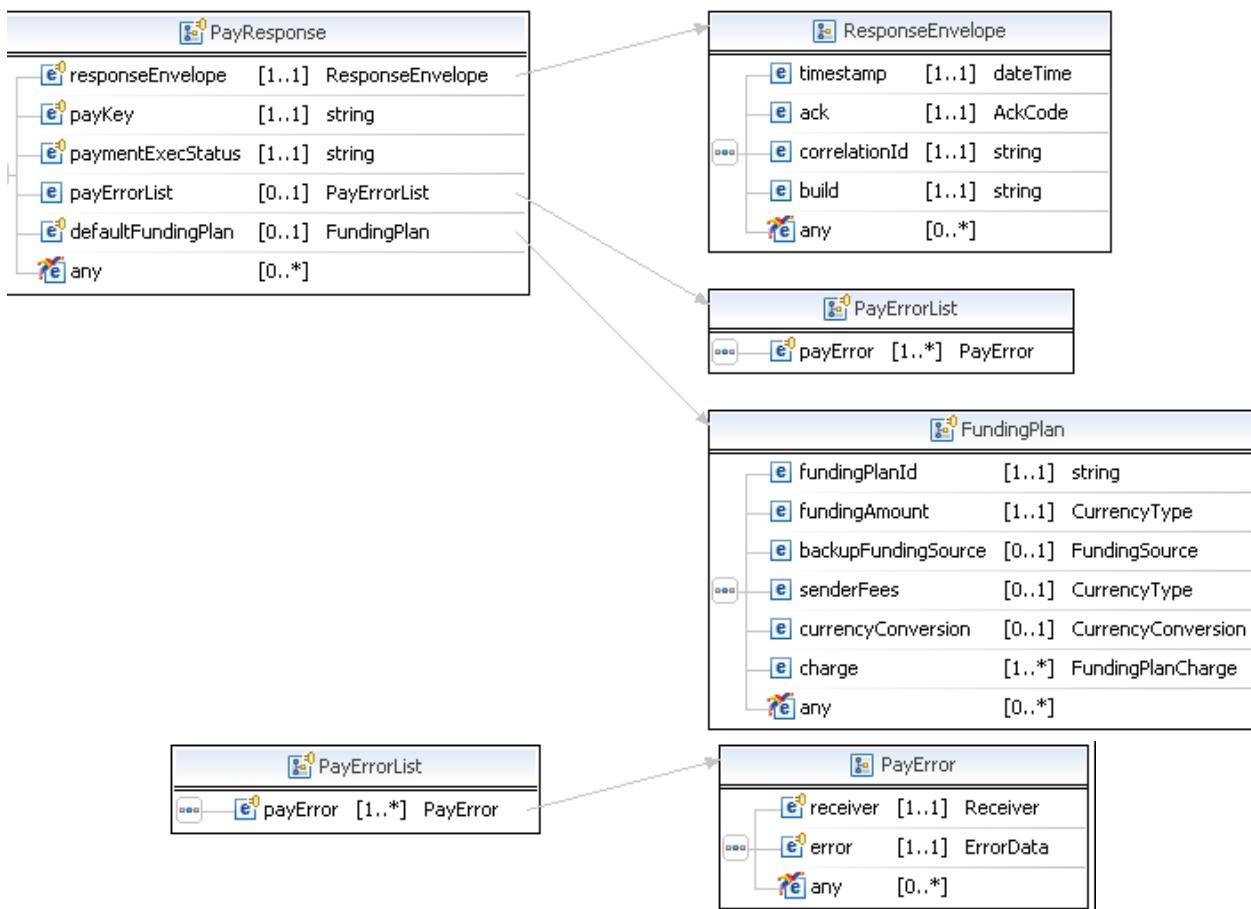
Field	Description
detailLevel	<p>common:DetailLevelCode <i>(Optional)</i> The level of detail required by the client application for component such as Item, Transaction. Possible values are: • ReturnAll – This value provides the maximum level of detail (default)</p>
errorLanguage	<p>xs:string <i>(Required)</i> The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported</p>

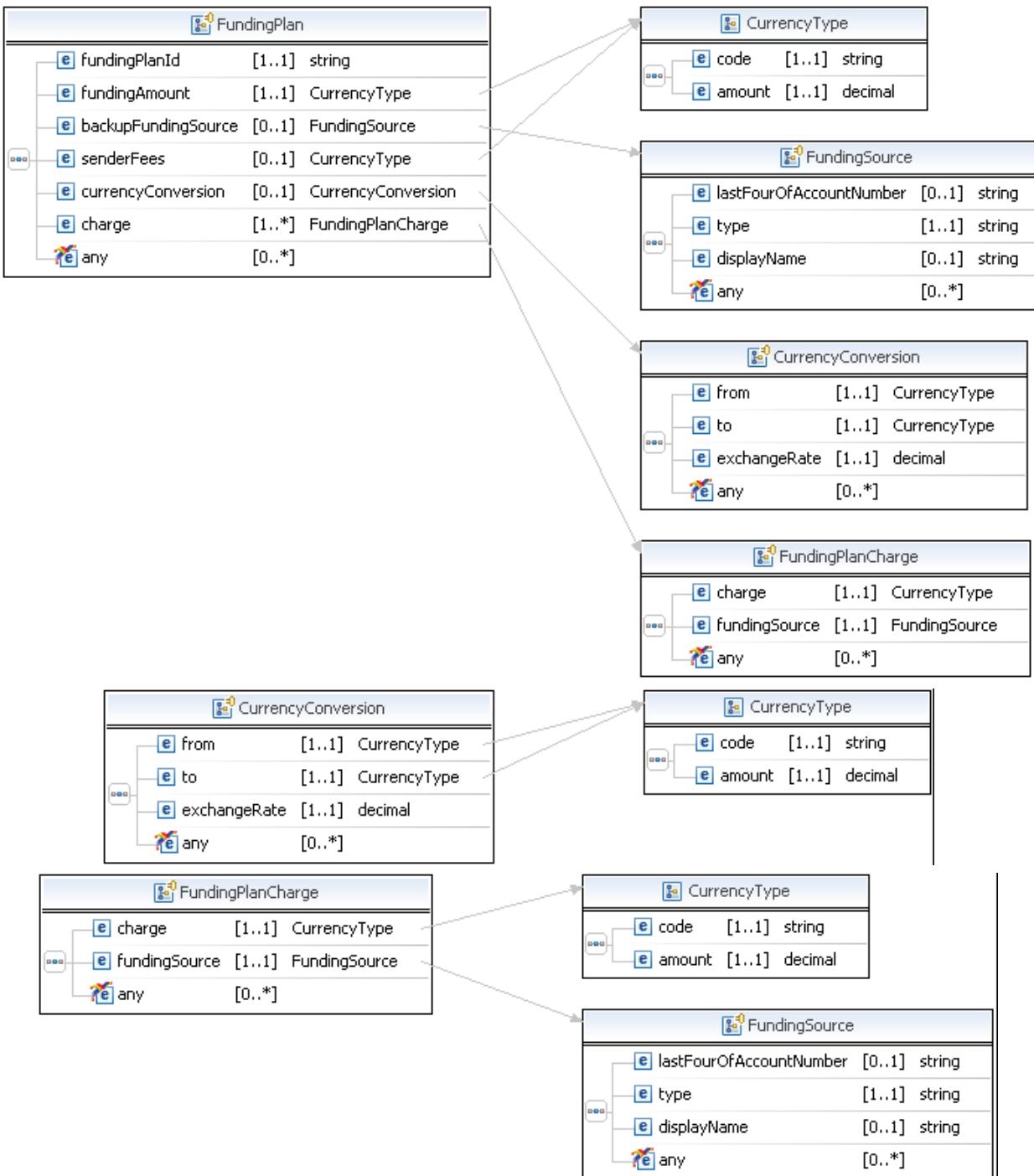
PayResponse Message

The PayResponse message contains a key that you can use to identify the payment and the payment's status.

Pay API Operation

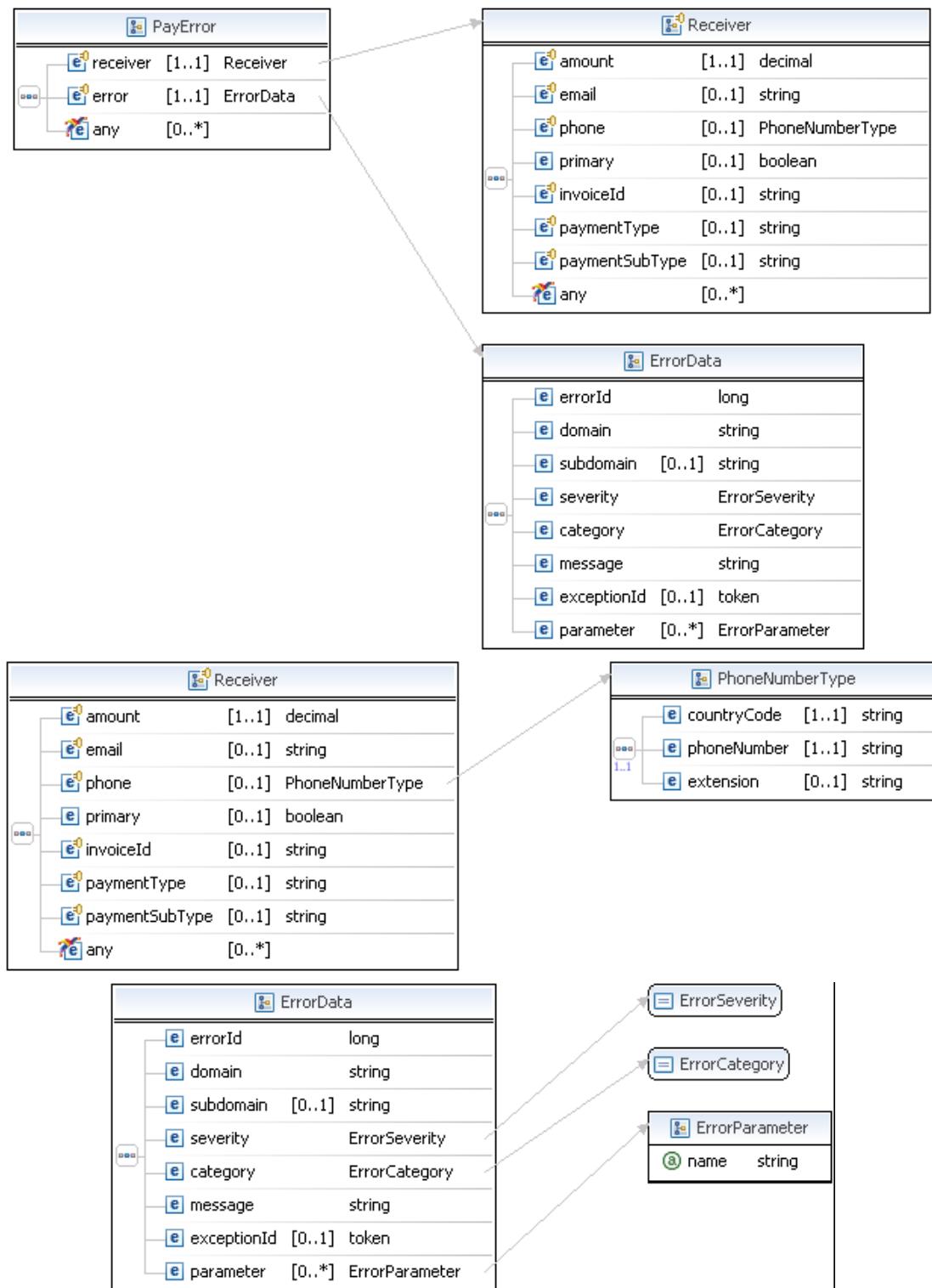
PayResponse Message

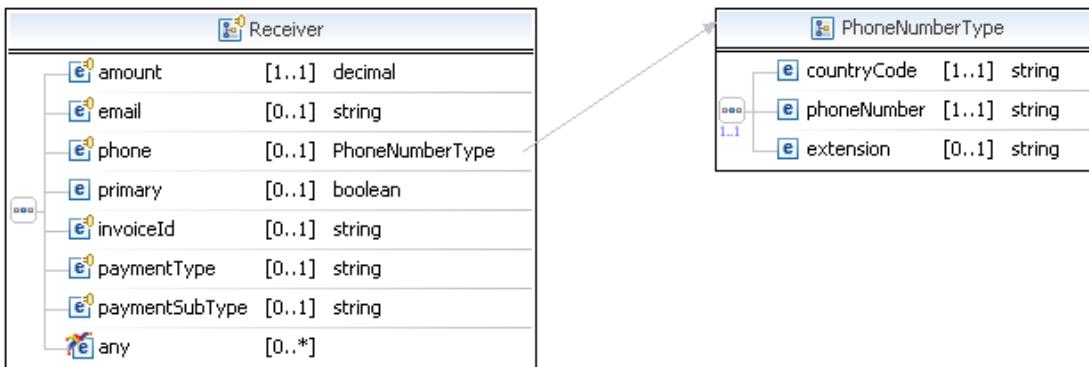




Pay API Operation

PayResponse Message





PayResponse Fields

Field	Description
payKey	<p>xs:string</p> <p>The pay key, which is a token you use in other Adaptive Payment APIs (such as the Refund Method) to identify this payment. The pay key is valid for 3 hours; the payment must be approved while the pay key is valid.</p>
payErrorList	<p>xs:string</p> <p>Information about why a payment failed.</p>
paymentExecStatus	<p>xs:string</p> <p>The status of the payment. Possible values are:</p> <ul style="list-style-type: none"> • CREATED – The payment request was received; funds will be transferred once the payment is approved • COMPLETED – The payment was successful • INCOMPLETE – Some transfers succeeded and some failed for a parallel payment or, for a delayed chained payment, secondary receivers have not been paid • ERROR – The payment failed and all attempted transfers failed or all completed transfers were successfully reversed • REVERSALERROR – One or more transfers failed when attempting to reverse a payment • PROCESSING – The payment is in progress • PENDING – The payment is awaiting processing
defaultFundingPlan	<p>ap:FundingPlan</p> <p>Default funding plan.</p>
responseEnvelope	<p>common:responseEnvelope</p> <p>Common response information, including a timestamp and the response acknowledgement status.</p>

Pay API Operation

PayResponse Message

PayErrorList Fields

Field	Description
payError	PayError indicates the error, if any, that resulted on an attempted payment to a receiver.

FundingPlan Fields

Field	Description
fundingPlanId	xs:string ID for this funding plan.
fundingAmount	common:CurrencyType Amount of available funding.
backupFundingSource	ap:FundingSource Backup funding source.
senderFees	common:CurrencyType Fees to be paid by the sender.
currencyConversion	ap:CurrencyConversion The currency conversion associated with the funding plan.
charge	ap:FundingPlanCharge The amount to be charged to this funding plan.

CurrencyType Fields

Field	Description
amount	xs:decimal The converted amount.

Field	Description
code	<p>xs:string</p> <p>The currency code for the converted amount. Possible values are:</p> <ul style="list-style-type: none"> • Australian Dollar – AUD • Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none"> • Canadian Dollar – CAD • Czech Koruna – CZK • Danish Krone – DKK • Euro – EUR • Hong Kong Dollar – HKD • Hungarian Forint – HUF • Israeli New Sheqel – ILS • Japanese Yen – JPY • Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none"> • Mexican Peso – MXN • Norwegian Krone – NOK • New Zealand Dollar – NZD • Philippine Peso – PHP • Polish Zloty – PLN • Pound Sterling – GBP • Singapore Dollar – SGD • Swedish Krona – SEK • Swiss Franc – CHF • Taiwan New Dollar – TWD • Thai Baht – THB • U.S. Dollar – USD

FundingSource Fields

Field	Description
lastFourOfAccountNumber	<p>xs:string</p> <p>Last 4 digits or characters in account number.</p>

Pay API Operation

PayResponse Message

Field	Description
type	<p>xs:string</p> <p>Type of funding source, which is one of the following values:.</p> <ul style="list-style-type: none">● UNDEFINED● BALANCE● BANK_INSTANT● BANK_DELAYED● CREDITCARD● DEBITCARD● ACCOUNTS_RECEIVABLE
displayName	<p>ap:FundingSource</p> <p>Display name of funding source.</p>

CurrencyConversion Fields

Field	Description
from	<p>ap:CurrencyType</p> <p>The currency to be converted.</p>
to	<p>ap:CurrencyType</p> <p>The currency resulting from the conversion.</p>
exchangeRate	<p>xs:decimal</p> <p>The exchange rate for the <code>from</code> currency to the <code>to</code> currency.</p>

FundingPlanCharge Fields

Field	Description
charge	<p>common:CurrencyType</p> <p>Amount charged to funding source.</p>
fundingSource	<p>ap:FundingSource</p> <p>Funding source being charged.</p>

PayError Fields

Field	Description
error	Detailed error information.

Field	Description
receiver	<p>ap:Receiver</p> <p>Receiver is the party where funds are transferred to. A primary receiver receives a payment directly from the sender in a chained split payment. A primary receiver should not be specified when making a single or parallel split payment.</p>

ErrorData Fields

Field	Description
category	<p>common:ErrorCategory</p> <p>The location where the error occurred.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • System – The system encountered errors; try again • Application – The application encountered errors; try again • Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	<p>common:ErrorParameter</p> <p>Represents contextual information about the error.</p>
severity	<p>common:ErrorSeverity</p> <p>The severity of the error encountered.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Error – Processing of the request was interrupted • Warning – Processing of the request was completed
subdomain	This field is not used.

Receiver Fields

Field	Description
amount	<p>xs:decimal</p> <p>Amount to be paid to the receiver.</p>
email	<p>xs:string</p> <p>Receiver's email address.</p> <p>Maximum length: 127 characters</p>

Pay API Operation

PayResponse Message

Field	Description
invoiceId	<p><code>xs:string</code></p> <p>The invoice number for the payment. This data in this field shows on the Transaction Details report.</p> <p>Maximum length: 127 characters</p>
paymentType	<p><code>xs:string</code></p> <p>The transaction type for the payment.</p> <p>Possible values are:</p> <ul style="list-style-type: none">• GOODS – This is a payment for non-digital goods• SERVICE – This is a payment for services (default)• PERSONAL – This is a person-to-person payment• CASHADVANCE – This is a person-to-person payment for a cash advance• DIGITALGOODS – This is a payment for digital goods
paymentSubType	<p><code>xs:string</code></p> <p>The transaction subtype for the payment.</p>
phone	<p><code>common:PhoneNumberType</code></p> <p>The receiver's phone number.</p>
primary	<p><code>xs:boolean</code></p> <p>Whether this receiver is the primary receiver. If this field is set to true, this is a chained payment. If this field shows false, this is a simple or parallel payment.</p> <p>Possible values are:</p> <ul style="list-style-type: none">• <code>true</code> – Primary receiver (chained payment)• <code>false</code> – Secondary receiver (simple/parallel payment)

PhoneNumberType Response Fields

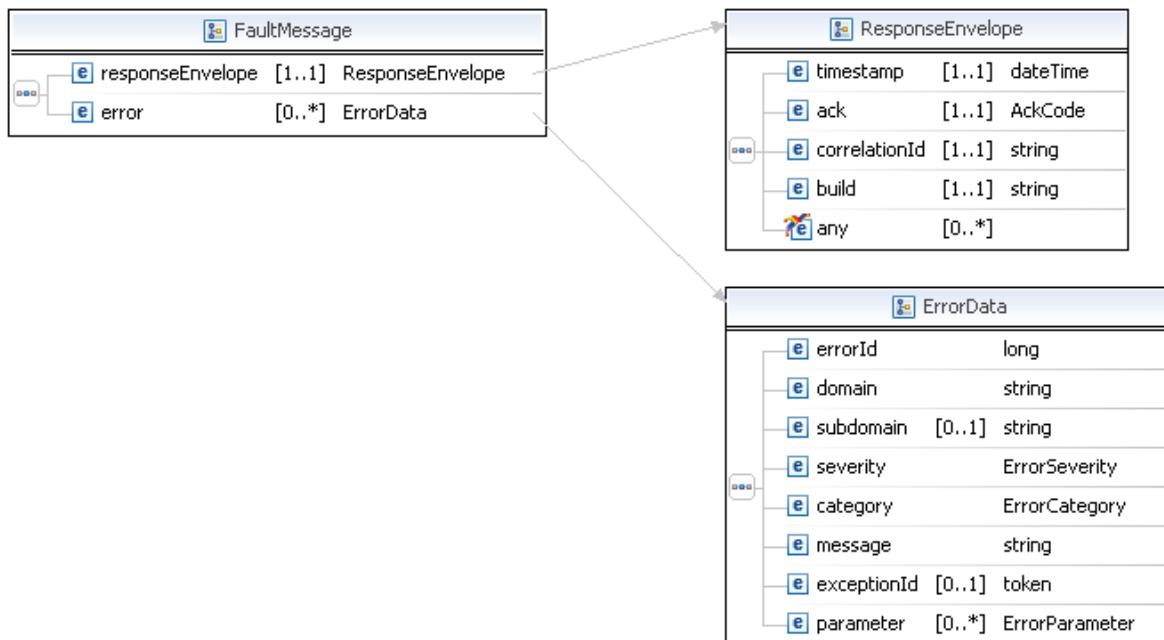
Field	Description
countryCode	<p><code>xs:string</code></p> <p>Telephone country code.</p>
phoneNumber	<p><code>xs:string</code></p> <p>Telephone number.</p>
extension	<p><code>xs:string</code></p> <p>Telephone extension.</p>

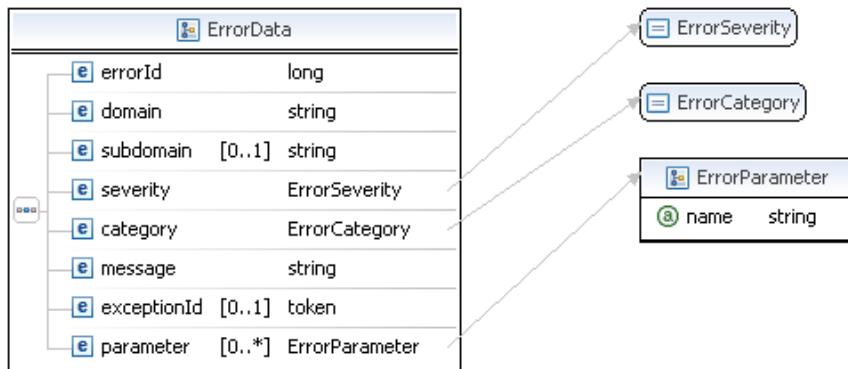
ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.





FaultMessage Fields

Field	Description
error	common:ErrorData Detailed error information.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	common:ErrorCategory The location where the error occurred. Possible values are: <ul style="list-style-type: none">• System – The system encountered errors; try again• Application – The application encountered errors; try again• Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	common:ErrorParameter Represents contextual information about the error.

Field	Description
severity	<p>common:ErrorSeverity</p> <p>The severity of the error encountered.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Error – Processing of the request was interrupted • Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	<p>common:AckCode</p> <p>Acknowledgement code. Possible values are:</p> <ul style="list-style-type: none"> • Success – Operation completed successfully • Failure – Operation failed • Warning – Warning • SuccessWithWarning – Operation completed successfully; however, there is a warning message • FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

Pay Examples Using NVP and CURL

These examples all use NVP for the data binding and CURL to deliver the HTTP request to the PayPal sandbox endpoint. Line breaks are provided for ease of reading; each CURL command is a single line and each request and response is a string without line breaks or extra whitespace.

Simple payment example

In this example, the sender makes a payment of \$100 to a PayPal-registered receiver. If you are the sender and the caller, the approval is implicit; otherwise, the sender must explicitly approve the payment.

The status of the request, which is identified in the paymentExecStatus field of the response, differs due to the kind of approval required. Implicit approval allows the request to be completed immediately. Explicit approval allows the request to be created; however, it is completed until the payment is approved.

Pay API Operation

Pay Examples Using NVP and CURL

NOTE: The sample code below uses the `insecure` setting to work around the certificate for testing in a sandbox environment. For actual implementations, you must specify the location of the certificate.

Request:

```
curl -s --insecure
-H "X-PAYPAL-SECURITY-USERID: api_username"
-H "X-PAYPAL-SECURITY-PASSWORD: api_password"
-H "X-PAYPAL-SECURITY-SIGNATURE: api_signature"
-H "X-PAYPAL-REQUEST-DATA-FORMAT: NV"
-H "X-PAYPAL-RESPONSE-DATA-FORMAT: NV"
-H "X-PAYPAL-APPLICATION-ID: app_id"
https://svcs.sandbox.paypal.com/AdaptivePayments/Pay -d
"requestEnvelope.errorLanguage=en_US
&actionType=PAY
&senderEmail=sender@domain
&receiverList.receiver(0).email=receiver@domain
&receiverList.receiver(0).amount=100.00
&currencyCode=USD
&feesPayer=EACHRECEIVER
&memo=Simple payment example.
&cancelUrl=http://your_cancel_url
&returnUrl=http://your_return_url
&ipnNotificationUrl=http://your_ipn_notification_url"
```

Response for an explicitly approved payment:

```
responseEnvelope.timestamp=2009-07-13T12%3A34%3A29.316-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=d615a365bed61
&responseEnvelope.build=DEV
&payKey=AP-3TY011106S4428730
&paymentExecStatus=CREATED
```

NOTE: You must redirect the sender to PayPal to complete the payment.

Response for an implicitly approved payment:

```
responseEnvelope.timestamp=2009-07-10T11%3A47%3A29.311-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=34e44c0bdbed6
&responseEnvelope.build=DEV
&payKey=AP-54224401WG093204T
&paymentExecStatus=COMPLETED
```

Parallel payment example

In this example, the sender makes a payment of \$100 to a PayPal-registered receiver and \$50 to another PayPal-registered receiver. If an error occurs, all funds are returned to the sender whether or not the funds were transferred to a receiver.

NOTE: The sample code below uses the `insecure` setting to work around the certificate for testing in a sandbox environment. For actual implementations, you must specify the location of the certificate.

Request:

```
curl -s --insecure
-H "X-PAYPAL-SECURITY-USERID: api_username"
-H "X-PAYPAL-SECURITY-PASSWORD: api_password"
-H "X-PAYPAL-SECURITY-SIGNATURE: api_signature"
-H "X-PAYPAL-REQUEST-DATA-FORMAT: NV"
-H "X-PAYPAL-RESPONSE-DATA-FORMAT: NV"
-H "X-PAYPAL-APPLICATION-ID: app_id"
https://svcs.sandbox.paypal.com/AdaptivePayments/Pay -d
?actionType=PAY
&cancelUrl=http://your_cancel_url
&currencyCode=USD
&feesPayer=EACHRECEIVER
&memo=Parallel-payment-example
&receiverList.receiver(0).amount=100.00
&receiverList.receiver(0).email=receiver1@domain
&receiverList.receiver(0).primary=false
&receiverList.receiver(1).amount=50.00
&receiverList.receiver(1).email=receiver2@domain
&receiverList.receiver(1).primary=false
&requestEnvelope.errorLanguage=en_US
&returnUrl=http://your_return_url
&reverseAllParallelPaymentsOnError=true
&senderEmail=sender@domain
```

Response:

Pay API Operation

Pay Examples Using NVP and CURL

```
responseEnvelope.timestamp=2009-07-13T12%3A34%3A29.316-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=d615a365bed61
&responseEnvelope.build=DEV&payKey=AP-1CB41255YH2406602
&paymentExecStatus=CREATED
```

Chained payment example

In this example, the sender makes a payment of \$100 to a PayPal-registered receiver who is identified as the primary receiver. This receiver sends \$50 of the \$100 to another PayPal-registered receiver.

Request:

```
curl -s --insecure
-H "X-PAYPAL-SECURITY-USERID: api_username"
-H "X-PAYPAL-SECURITY-PASSWORD: api_password"
-H "X-PAYPAL-SECURITY-SIGNATURE: api_signature"
-H "X-PAYPAL-REQUEST-DATA-FORMAT: NV"
-H "X-PAYPAL-RESPONSE-DATA-FORMAT: NV"
-H "X-PAYPAL-APPLICATION-ID: app_id"
https://svcs.sandbox.paypal.com/AdaptivePayments/Pay -d
?actionType=PAY
&cancelUrl=http://your_cancel_url
&clientDetails.applicationId=fitness
&clientDetails.ipAddress=127.0.0.1
&currencyCode=USD
&feesPayer=EACHRECEIVER
&memo=Parallel-payment-example
&receiverList.receiver(0).amount=100.00
&receiverList.receiver(0).email=receiver1@domain
&receiverList.receiver(0).primary=true
&receiverList.receiver(1).amount=50.00
&receiverList.receiver(1).email=receiver2@domain
&receiverList.receiver(1).primary=false
&requestEnvelope.errorLanguage=en_US
&returnUrl=http://your_return_url
&senderEmail=sender@domain
```

Response:

```
responseEnvelope.timestamp=2009-07-13T12%3A34%3A29.316-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=d615a365bed61
&responseEnvelope.build=DEV
&payKey=AP-7AF63289GN043650D
&paymentExecStatus=CREATED
```

Preapproved payment example

In this example, the sender has a valid preapproval agreement with you and makes a payment of \$100 to a PayPal-registered receiver and \$50 to another PayPal-registered receiver. The payment is completed without the sender logging in to paypal.com.

NOTE: The sample code below uses the `insecure` setting to work around the certificate for testing in a sandbox environment. For actual implementations, you must specify the location of the certificate.

Request:

```
curl -s --insecure
-H "X-PAYPAL-SECURITY-USERID: api_username"
-H "X-PAYPAL-SECURITY-PASSWORD: api_password"
-H "X-PAYPAL-SECURITY-SIGNATURE: api_signature"
-H "X-PAYPAL-REQUEST-DATA-FORMAT: NV"
-H "X-PAYPAL-RESPONSE-DATA-FORMAT: NV"
-H "X-PAYPAL-APPLICATION-ID: app_id"
https://svcs.sandbox.paypal.com/AdaptivePayments/Pay -d
?actionType=PAY
&cancelUrl=http://your_cancel_url
&currencyCode=USD
&feesPayer=EACHRECEIVER
&memo=Preapproval-payment-example
&preapprovalKey=PA-9JR04288NR0519129
&receiverList.receiver(0).amount=100.00
&receiverList.receiver(0).email=receiver1@domain
&receiverList.receiver(0).primary=false
&receiverList.receiver(1).amount=50.00
&receiverList.receiver(1).email=receiver2@domain
&receiverList.receiver(1).primary=false
&requestEnvelope.errorLanguage=en_US
&returnUrl=http://your_return_url
&reverseAllParallelPaymentsOnError=true
&senderEmail=sender@domain
```

Response:

Pay API Operation

Pay Examples Using NVP and CURL

```
responseEnvelope.timestamp=2009-07-13T12%3A34%3A29.316-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=d615a365bed61
&responseEnvelope.build=DEV
&payKey=AP-42706441J12795911
&paymentExecStatus=COMPLETED
```

Digital goods payment

In this example, the sender makes a payment of \$1 to a PayPal-registered receiver. The sender must explicitly approve the payment.

NOTE: The sample code below uses the `insecure` setting to work around the certificate for testing in a sandbox environment. For actual implementations, you must specify the location of the certificate.

Request:

```
curl -s --insecure
-H "X-PAYPAL-SECURITY-USERID: api_username"
-H "X-PAYPAL-SECURITY-PASSWORD: api_password"
-H "X-PAYPAL-SECURITY-SIGNATURE: api_signature"
-H "X-PAYPAL-REQUEST-DATA-FORMAT: NV"
-H "X-PAYPAL-RESPONSE-DATA-FORMAT: NV"
-H "X-PAYPAL-APPLICATION-ID: app_id"
https://svcs.sandbox.paypal.com/AdaptivePayments/Pay -d
"requestEnvelope.errorLanguage=en_US
&actionType=PAY
&receiverList.receiver(0).amount=1.00
&currencyCode=USD
&feesPayer=EACHRECEIVER
&memo=Simple payment example.
&cancelUrl=http://your_cancel_url
&returnUrl=http://your_return_url
&ipnNotificationUrl=http://your_ipn_notification_url
&receiverList.receiver(0).paymentType=DIGITALGOODS"
```

Response to a payment for digital goods:

```
responseEnvelope.timestamp=2009-07-13T12%3A34%3A29.316-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=d615a365bed61
&responseEnvelope.build=DEV
&payKey=AP-3TY011106S4428730
&paymentExecStatus=CREATED
```

NOTE: You must redirect the sender to the following PayPal URL to complete the payment for digital goods:

```
https://paypal.com/webapps/adaptivepayment/flow/pay?paykey=....
```

Pay Errors

Code	Message	Additional Information
500000	There is a system error	
520002	Internal error	
520003	User name/password is incorrect	
520003	<ul style="list-style-type: none"> • The caller's credential doesn't have permission for the fee payer <value> • The caller's credential doesn't have permission for the payment type <value> • Authentication failed. API credentials are incorrect. 	Since 1.6.0
520005	Merchant account is locked	
520006	This call is not defined in the database	
529038	There was an error while making this payment	
539012	The preapproval key has not been authorized yet	
539041	The email account is based in a country that is not enabled to receive payments	
539043	The email account is based in a country that is not enabled to send payments	
540031	You don't have permission to cancel this preapproval	
550001	<ul style="list-style-type: none"> • You are not allowed to confirm the preapproval for this key • User is not allowed to perform this action. 	Since 1.6.0
559044	Account setting on the receiver prohibited the payment	
560027	The argument value is unsupported	
569000	Split payments are not supported at this time	
569013	The preapproval key has been canceled	
569016	Preapproval PIN functionality is not enabled	
569017	The preapproval key has been suspended	
569018	Preapproved payments have been disabled	
569019	The preapproval has been suspended due to too many PIN failures	
569042	The email account is not confirmed by PayPal	
579007	The maximum number of receivers is <number>	
579010	If a preapproval key is specified, the sender's email address must be, too	

Code	Message	Additional Information
579014	The preapproval key specifies a different sender than the payment request	
579017	The amount for the primary receiver must be greater than or equal to the total of other chained receiver amounts	
579024	The preapproval key cannot be used before the start date or after the end date	
579025	The preapproval key cannot be used on this weekday	
579026	The preapproval key cannot be used on this day of the month	
579027	The preapproval key specifies a different currency than the payment request	
579028	The payment amount exceeds the maximum amount per payment	
579030	The number of payments made this period exceeds the maximum number of payments per period	
579031	The total amount of all payments exceeds the maximum total amount for all payments	
579033	The sender and each receiver must have different accounts	
579040	The receivers cannot belong to the same PayPal account	
579042	The tracking ID already exists and cannot be duplicated	
579045	The email account exceeds the receiving limit	
579047	The email account exceeds the purse limit	
579048	The email account exceeds the sending limit	
580001	Invalid request	
580001	<ul style="list-style-type: none"> • Invalid request: both email and phone cannot be set <value> <value> <value> • Invalid request: neither email nor phone is set <value> • To use currency <value>, currency must be held by sender and all receivers • Invalid request: More than one field cannot be used to specify a sender • Invalid request: <value> 	Since 1.6.0
580022	Your request was invalid. Check the parameter of the error to see which is invalid	

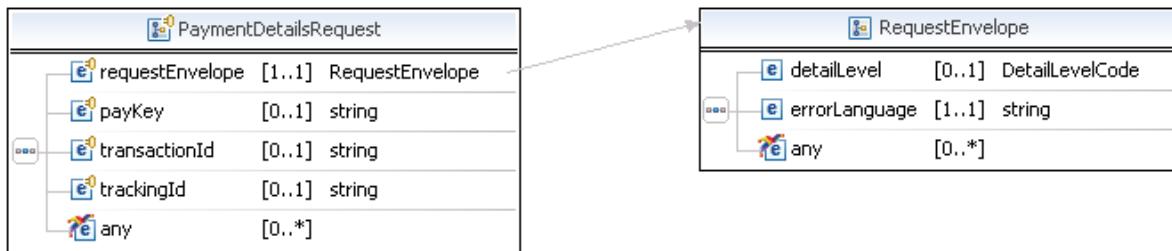
Code	Message	Additional Information
580022	<ul style="list-style-type: none"> • Invalid request parameter: <value> • Invalid Request Parameter: institutionId with value <value> • The institution id <value> is not activated • The clientDetails parameter is missing • Invalid request parameter: payKey with value <value> • Invalid Phone Country Code <value> • Invalid Country Code <value> for Receiver • Invalid Receiver Phone Type <value> <value> <value> • Invalid Phone Format <value> <value> • Invalid Phone Number <value> <value> • Invalid Receiver Phone Number <value> <value> • Invalid Area or Exchange Code <value> <value> • <value> has already been set and cannot be set multiple times • Invalid request parameter: <value> with value <value> • Invalid request parameter: <value> <value> is invalid • Invalid request parameter: <value> <value> is duplicated • Invalid request parameter: <value> <value> cannot be negative or zero 	Since 1.6.0
580023	Invalid request	
580023	<ul style="list-style-type: none"> • The fee payer <value> cannot be used if a primary receiver is specified • The fee payer <value> can only be used if a primary receiver is specified • The fee payer <value> must have exactly 2 receivers • The reverseAllParallelPaymentOnError field can be true only if a primary receiver is not specified • The fee payer <value> cannot be used if a preapproval key is specified • Invalid request: FundingType cannot be specified when PaymentSubType is specified • Your Application is not approved to use the PaymentSubType: <value> • Your Application is not approved to use the PaymentType: <value> • Invalid request parameter: PaymentType should be the same for all receivers <value> • Inconsistent request parameter for digitalgoods • Invalid request parameter: PaymentSubType should be the same for all receivers <value> • Invalid Request: RECEIVABLE cannot be used with other funding types • For chained payments, receivers must have premier or business accounts 	Since 1.6.0
580027	The argument is unsupported	

Code	Message	Additional Information
580027	<ul style="list-style-type: none">• Currently the system does not accept the currency <value>• Unsupported country code <value>• Currently the system does not accept the currency <value>• The parameter is not supported	Since 1.6.0
580028	The URL <value> is malformed	
580029	Invalid request	
589009	This payment cannot be processed because no payment source is available	
589019	The preapproval key is invalid	
589023	If a fractional amount is rounded due to currency conversion, funds could be lost	
589039	The email address is invalid. It may not be registered in PayPal's system yet	

Use the PaymentDetails API operation to obtain information about a payment. You can identify the payment by your tracking ID, the PayPal transaction ID in an IPN message, or the pay key associated with the payment.

PaymentDetailsRequest Message

The PaymentDetailsRequest message contains the identifier used to retrieve information about the payment.



PaymentDetailsRequest Fields

The `payKey`, `transactionId`, and `trackingId` fields identify the payment. Only one is required to identify the payment; however, if more than one is specified, all fields must identify the same payment.

Field	Description
<code>payKey</code>	<code>xs:string</code> <i>(Optional)</i> The pay key that identifies the payment for which you want to retrieve details. This is the pay key returned in the <code>PayResponse</code> message.
<code>requestEnvelope</code>	<code>common:RequestEnvelope</code> <i>(Required)</i> Information common to each API operation, such as the language in which an error message is returned.
<code>transactionId</code>	<code>xs:string</code> <i>(Optional)</i> The PayPal transaction ID associated with the payment. The IPN message associated with the payment contains the transaction ID.

PaymentDetails API Operation

PaymentDetailsResponse Message

Field	Description
trackingId	<p>xs:string</p> <p>(Optional) The tracking ID that was specified for this payment in the PayRequest message.</p> <p>Maximum length: 127 characters</p>

RequestEnvelope Fields

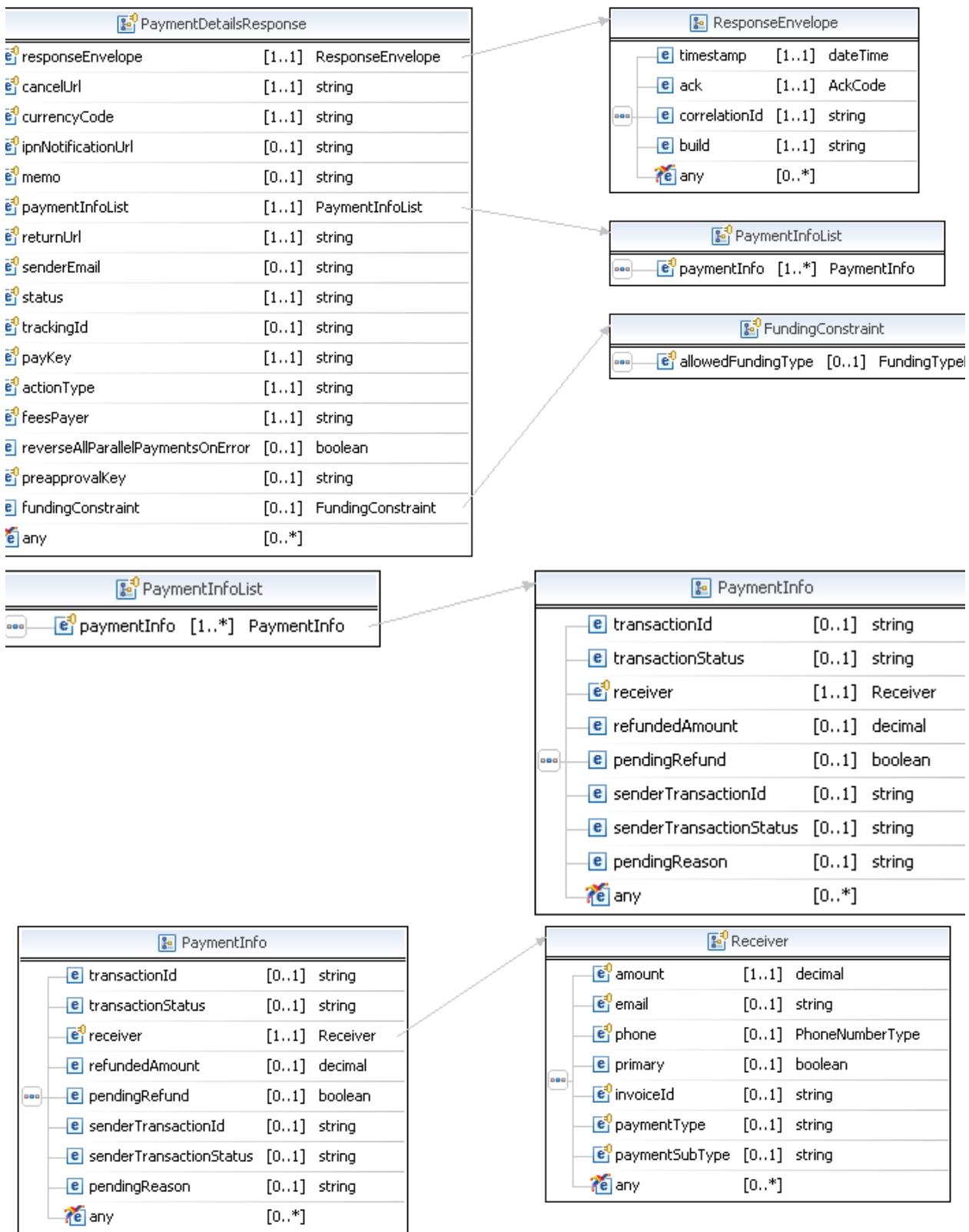
Field	Description
detailLevel	<p>common:DetailLevelCode</p> <p>(Optional) The level of detail required by the client application for component such as Item, Transaction. Possible values are:</p> <ul style="list-style-type: none">• ReturnAll – This value provides the maximum level of detail (default)
errorLanguage	<p>xs:string</p> <p>(Required) The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported</p>

PaymentDetailsResponse Message

The PaymentDetailsResponse message contains information about the payment.

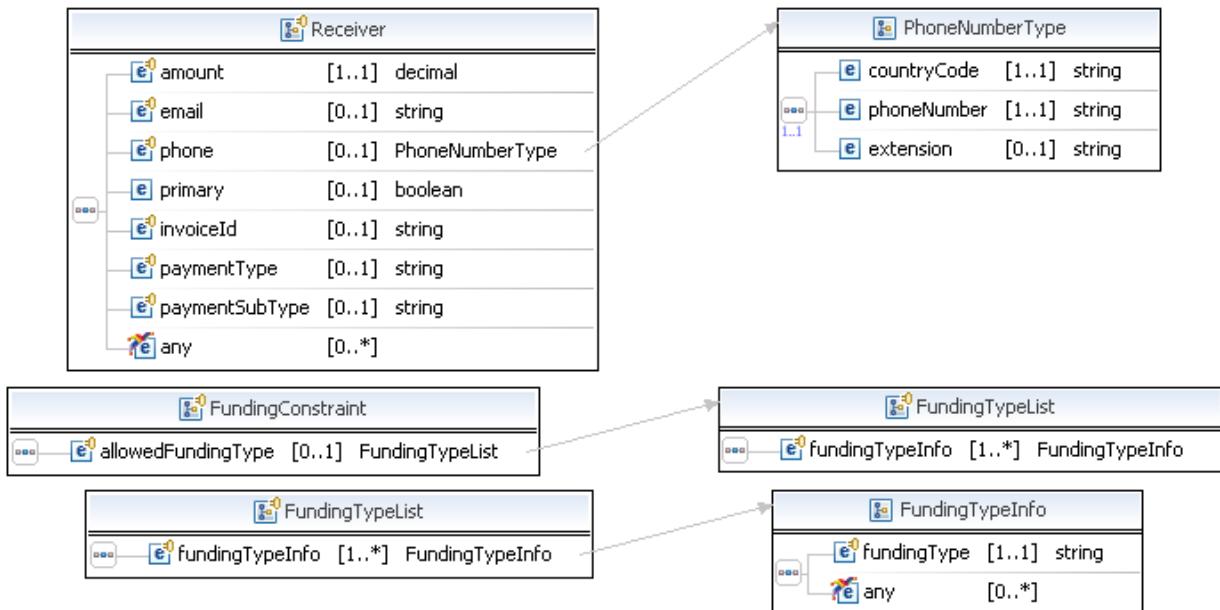
PaymentDetails API Operation

PaymentDetailsResponse Message



PaymentDetails API Operation

PaymentDetailsResponse Message



PaymentDetailsResponse Fields

Field	Description
<code>actionType</code>	<code>xs:string</code> Whether the Pay request is set up to create a payment request with the SetPaymentOptions request, and then fulfill the payment with the ExecutePayment request. Possible values are: <ul style="list-style-type: none">• PAY – Use this option if you are not using the Pay request in combination with ExecutePayment.• CREATE – Use this option to set up the payment instructions with SetPaymentOptions and then execute the payment at a later time with the ExecutePayment.• PAY_PRIMARY – For chained payments only, specify this value to delay payments to the secondary receivers; only the payment to the primary receiver is processed.
<code>cancelUrl</code>	<code>xs:string</code> The URL to which the sender's browser is redirected if the sender cancels the approval for a payment on paypal.com. Use the pay key to identify the payment as follows: <code>payKey=\${payKey}</code> .

Field	Description
currencyCode	<p>xs:string</p> <p>The currency code. Possible values are:</p> <ul style="list-style-type: none"> • Australian Dollar – AUD • Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none"> • Canadian Dollar – CAD • Czech Koruna – CZK • Danish Krone – DKK • Euro – EUR • Hong Kong Dollar – HKD • Hungarian Forint – HUF • Israeli New Sheqel – ILS • Japanese Yen – JPY • Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none"> • Mexican Peso – MXN • Norwegian Krone – NOK • New Zealand Dollar – NZD • Philippine Peso – PHP • Polish Zloty – PLN • Pound Sterling – GBP • Singapore Dollar – SGD • Swedish Krona – SEK • Swiss Franc – CHF • Taiwan New Dollar – TWD • Thai Baht – THB • U.S. Dollar – USD
feesPayer	<p>xs:string</p> <p>The payer of PayPal fees. Possible values are:</p> <ul style="list-style-type: none"> • SENDER – Sender pays all fees (for personal, implicit simple/parallel payments; do not use for chained or unilateral payments) • PRIMARYRECEIVER – Primary receiver pays all fees (chained payments only) • EACHRECEIVER – Each receiver pays their own fee (default, personal and unilateral payments) • SECONDARYONLY – Secondary receivers pay all fees (use only for chained payments with one secondary receiver)

PaymentDetails API Operation

PaymentDetailsResponse Message

Field	Description
fundingtypeList	ap:FundingTypeList Specifies a list of allowed funding selections for the payment. This is a list of funding selections that can be combined in any order to allow payments to use the indicated funding type. If this Parameter is omitted, the payment can be funded by any funding type that is supported for Adaptive Payments.
ipnNotificationUrl	xs:string The URL to which all IPN messages for this payment are sent.
memo	xs:string A note associated with the payment.
payKey	xs:string The pay key that identifies this payment. This is a token that is assigned by the Pay API after a PayRequest message is received and can be used in other Adaptive Payments Methods, as well as the cancelURL and returnUrl to identify this payment. The pay key is valid for 3 hours .
paymentInfoList	ap:PaymentInfoList Represents the payment attempt made to a receiver of a PayRequest. If the execution of the payment has not yet completed, there are no transaction details returned.
preapprovalKey	xs:string The preapproval key that identifies the preapproval for this payment. This key can be used in other Adaptive Payment APIs to identify the preapproval.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.
returnUrl	xs:string The URL to which the sender's browser is redirected after approving a payment on paypal.com. Use the pay key to identify the payment as follows: payKey=\${payKey}.
reverseAllParallelPaymentsOnError	xs:string Whether to reverse parallel payments. Possible values are: <ul style="list-style-type: none">• true – Each parallel payment is reversed if an error occurs• false – Each parallel payment is not reversed if an error occurs
senderEmail	xs:string The sender's email address.

Field	Description
status	<p>xs:string</p> <p>The status of the payment. Possible values are:</p> <ul style="list-style-type: none"> • CREATED – The payment request was received; funds will be transferred once the payment is approved • COMPLETED – The payment was successful • INCOMPLETE – Some transfers succeeded and some failed for a parallel payment or, for a delayed chained payment, secondary receivers have not been paid • ERROR – The payment failed and all attempted transfers failed or all completed transfers were successfully reversed • REVERSALERROR – One or more transfers failed when attempting to reverse a payment • PROCESSING – The payment is in progress • PENDING – The payment is awaiting processing
trackingId	<p>xs:string</p> <p>The tracking ID that was specified for this payment in the PaymentDetailsRequest message.</p>

FundingTypeList Fields

Field	Description
fundingTypeInfo	<p>ap:FundingTypeInfo</p> <p>Specifies a list of allowed funding selections for the payment. This is a list of funding selections that can be combined in any order to allow payments to use the indicated funding type. If this field is omitted, the payment can be funded by any funding type that is supported for Adaptive Payments.</p>

FundingTypeInfo Fields

Field	Description
fundingType	<p>Specifies a list of allowed funding selections for the payment. This is a list of funding selections that can be combined in any order to allow payments to use the indicated funding type. If this field is omitted, the payment can be funded by any funding type that is supported for Adaptive Payments.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • ECHECK – Electronic check • BALANCE – PayPal account balance • CREDITCARD – Credit card

PaymentDetails API Operation

PaymentDetailsResponse Message

PaymentInfoList Fields

Fields	Description
paymentInfo	ap:PaymentInfo Represents the payment attempt made to a receiver of a PayRequest. If the execution of the payment has not yet completed, there are no transaction details returned.

PaymentInfo Fields

Field	Description
pendingReason	xs:string The reason why a payment is pending, the payment has a pending refund. Possible values are: <ul style="list-style-type: none">• ADDRESS_CONFIRMATION – The payment is pending because the sender does not have a confirmed address on file and receiver's Payment Receiving Preferences is set for manually accepting payments or denying each of these payments.• ECHECK – The payment is pending because it was made by an eCheck that has not yet cleared.• INTERNATIONAL – The payment is pending because the receiver holds a non-U.S. account and does not have a withdrawal mechanism. The receiver must manually accept or deny this payment from the Account Overview.• MULTI_CURRENCY – The receiver does not have a balance in the currency sent, and does not have the Payment Receiving Preferences set to automatically convert and accept this payment. Receiver must manually accept or deny this payment from the Account Overview.• UNILATERAL – The payment is pending because it was made to an email address that is not yet registered or confirmed.• UPGRADE – The payment is pending because it was made via credit card and the receiver must upgrade the account to a Business account or Premier status to receive the funds. It can also mean that receiver has reached the monthly limit for transactions on the account.• VERIFY – The payment is pending because the receiver is not yet verified.• RISK – The payment is pending while it is being reviewed by PayPal for risk.• OTHER – The payment is pending for a reason other than those listed above. For more information, contact PayPal Customer Service.
pendingRefund	xs:boolean Whether the payment has a pending refund. Possible values are: <ul style="list-style-type: none">• true – The payment has a pending refund.• false – The payment does not have a refund pending.

Field	Description
receiver	<p>ap:Receiver</p> <p>The receiver of the payment.</p>
refundedAmount	<p>xs:decimal</p> <p>The total amount refunded for this payment since the payment occurred.</p>
senderTransactionID	<p>xs:string</p> <p>The transaction identification number of the sender of the payment.</p>
senderTransactionStatus	<p>xs:string</p> <p>The transaction status of the payment.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • SUCCESS – The sender's transaction has completed • PENDING – The transaction is awaiting further processing • CREATED – The payment request was received; funds will be transferred once approval is received • PARTIALLY_REFUNDED – Transaction was partially refunded • DENIED – The transaction was rejected by the receiver • PROCESSING – The transaction is in progress • REVERSED – The payment was returned to the sender • REFUNDED – The payment was refunded • FAILED – The payment failed
transactionId	<p>xs:string</p> <p>PayPal transaction ID for this payment to the specified receiver.</p>
transactionStatus	<p>xs:string</p> <p>PayPal transaction status.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • SUCCESS – The sender's transaction has completed • PENDING – The transaction is awaiting further processing • CREATED – The payment request was received; funds will be transferred once approval is received • PARTIALLY_REFUNDED – Transaction was partially refunded • DENIED – The transaction was rejected by the receiver • PROCESSING – The transaction is in progress • REVERSED – The payment was returned to the sender • REFUNDED – The payment was refunded • FAILED – The payment failed

Receiver Fields

Field	Description
amount	<p>xs:decimal</p> <p>Amount to be paid to the receiver.</p>

PaymentDetails API Operation

PaymentDetailsResponse Message

Field	Description
email	<code>xs:string</code> Receiver's email address. Maximum length: 127 characters
invoiceId	<code>xs:string</code> The invoice number for the payment. This data in this field shows on the Transaction Details report. Maximum length: 127 characters
paymentType	<code>xs:string</code> The transaction type for the payment. Possible values are: <ul style="list-style-type: none">● GOODS – This is a payment for non-digital goods● SERVICE – This is a payment for services (default)● PERSONAL – This is a person-to-person payment● CASHADVANCE – This is a person-to-person payment for a cash advance● DIGITALGOODS – This is a payment for digital goods
paymentSubType	<code>xs:string</code> The transaction subtype for the payment.
phone	<code>common:PhoneNumberType</code> The receiver's phone number.
primary	<code>xs:boolean</code> Whether this receiver is the primary receiver. If this field is set to true, this is a chained payment. If this field shows false, this is a simple or parallel payment. Possible values are: <ul style="list-style-type: none">● true – Primary receiver (chained payment)● false – Secondary receiver (simple/parallel payment)

PhoneNumberType Response Fields

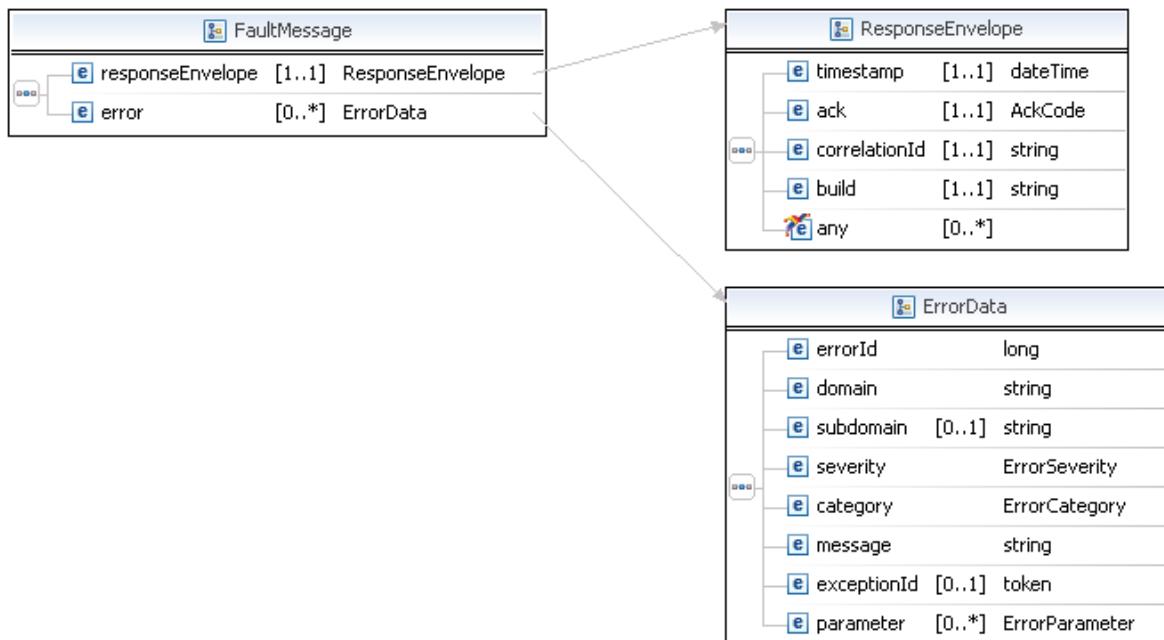
Field	Description
countryCode	<code>xs:string</code> Telephone country code.
phoneNumber	<code>xs:string</code> Telephone number.
extension	<code>xs:string</code> Telephone extension.

ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

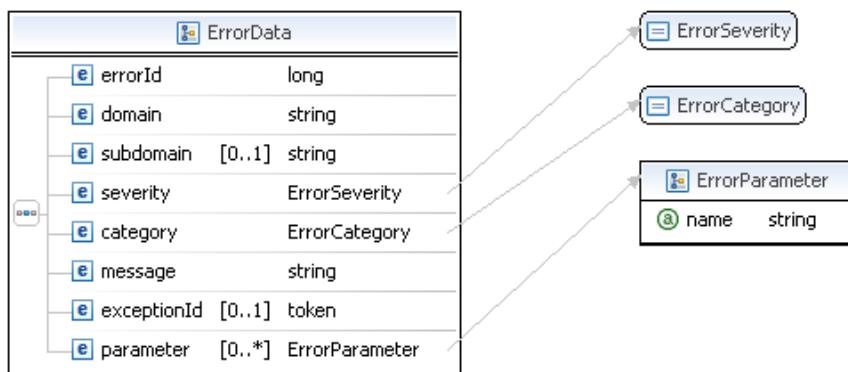
PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.



PaymentDetails API Operation

PPFault Message



FaultMessage Fields

Field	Description
error	common:ErrorData Detailed error information.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	common:ErrorCategory The location where the error occurred. Possible values are: <ul style="list-style-type: none">• System – The system encountered errors; try again• Application – The application encountered errors; try again• Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	common:ErrorParameter Represents contextual information about the error.

Field	Description
severity	<p>common:ErrorSeverity</p> <p>The severity of the error encountered.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Error – Processing of the request was interrupted • Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	<p>common:AckCode</p> <p>Acknowledgement code. Possible values are:</p> <ul style="list-style-type: none"> • Success – Operation completed successfully • Failure – Operation failed • Warning – Warning • SuccessWithWarning – Operation completed successfully; however, there is a warning message • FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

PaymentDetails Examples Using NVP and CURL

These examples use NVP for the data binding and CURL to deliver the HTTP request to the PayPal sandbox endpoint. Line breaks are provided for ease of reading; each CURL command is a single line and each request and response is a string without line breaks or extra whitespace.

Using the payment key to obtain payment details

In this example, the caller of the Pay API operation specifies the payment key to obtain information about the entire payment:

NOTE: The sample code below uses the `insecure` setting to work around the certificate for testing in a sandbox environment. For actual implementations, you must specify the location of the certificate.

Request:

PaymentDetails API Operation

Payment Details Errors

```
curl -s --insecure
-H "X-PAYPAL-SECURITY-USERID: api_username"
-H "X-PAYPAL-SECURITY-PASSWORD: api_password"
-H "X-PAYPAL-SECURITY-SIGNATURE: api_signature"
-H "X-PAYPAL-REQUEST-DATA-FORMAT: NV"
-H "X-PAYPAL-RESPONSE-DATA-FORMAT: NV"
-H "X-PAYPAL-APPLICATION-ID: app_id"
https://svcs.sandbox.paypal.com/AdaptivePayments/PaymentDetails -d
"payKey=AP-3TY011106S4428730
&requestEnvelope.errorLanguage=en_US"
```

Response:

```
responseEnvelope.timestamp=2009-08-14T09%3A00%3A37.748-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=7967b2d03745a
&responseEnvelope.build=DEV
&cancelUrl=your_cancel_url
&currencyCode=USD
&logDefaultShippingAddress=false
&memo=Simple payment example.
&paymentInfoList.paymentInfo(0).receiver.amount=100.00
&paymentInfoList.paymentInfo(0).receiver.email=receiver@domain
&paymentInfoList.paymentInfo(0).receiver.primary=false
&returnUrl=your_return_url
&senderEmail=sender@domain
&status=CREATED
&payKey=AP-3TY011106S4428730
&actionType=PAY
&feesPayer=EACHRECEIVER
```

Payment Details Errors

Code	Message	Additional Information
500000	There is a system error	
520002	Internal error	
520003	User name/password is incorrect	
520003	Authentication failed. API credentials are incorrect.	Since 1.6.0
520006	This call is not defined in the database	
540031	You do not have permission to get these payment details	Since 1.6.0
579049	The payment request you are looking for was not found in our system	

Code	Message	Additional Information
579051	The tracking ID is invalid	
580001	Invalid request	
580022	Invalid request parameter: payKey with value <value>	
580029	Invalid request	
589052	The transaction ID is invalid	

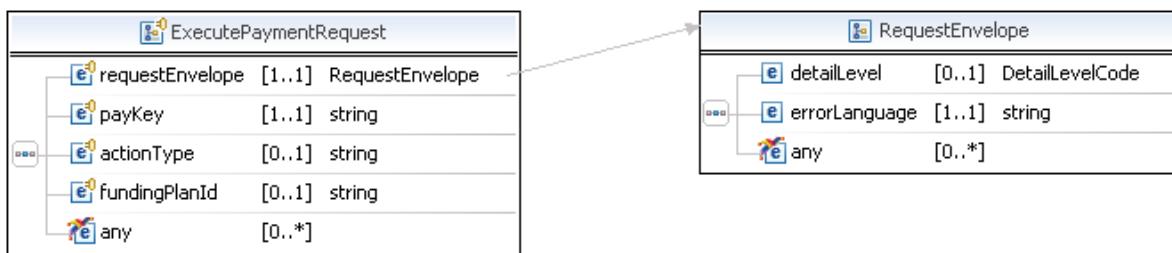
5

ExecutePayment API Operation

The ExecutePayment API operation lets you execute a payment set up with the Pay API operation with the `actionType` CREATE. To pay receivers identified in the Pay call, set the pay key from the PayResponse message in the ExecutePaymentRequest message.

ExecutePaymentRequest Message

The ExecutePaymentRequest message lets you execute a payment requested using the Pay API operation.



ExecutePaymentRequest Fields

Field	Description
payKey	<code>xs:string</code> <i>(Optional)</i> The pay key that identifies the payment to be executed. This is the pay key returned in the PayResponse message.
fundingPlanId	<code>xs:string</code> <i>(Optional)</i> The ID of the funding plan from which to make this payment.
requestEnvelope	<code>common:requestEnvelope</code> Information common to each API operation, such as the language in which an error message is returned.

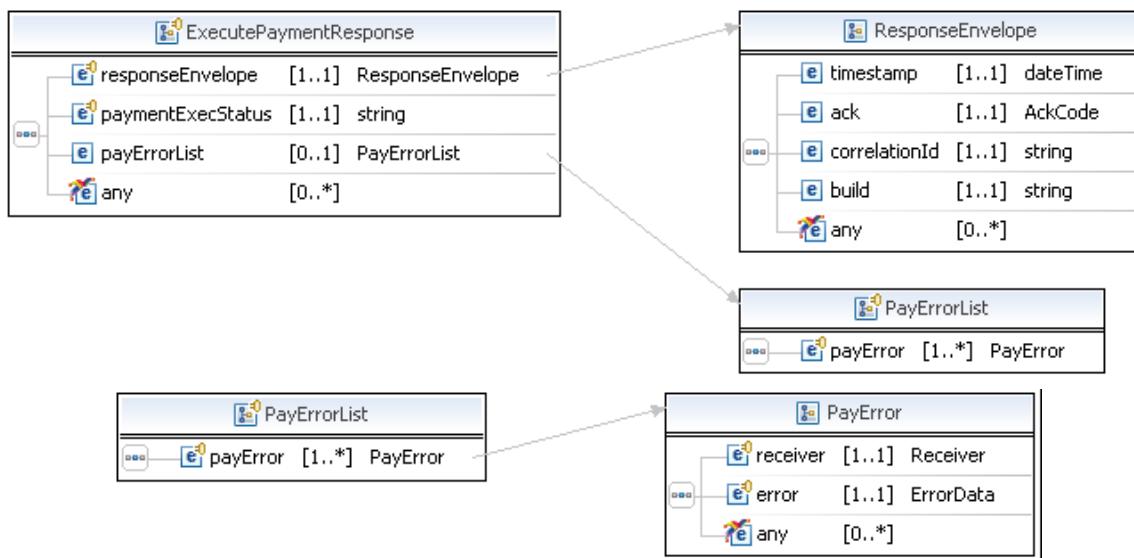
ExecutePayment API Operation

ExecutePaymentResponse Message

RequestEnvelope Fields

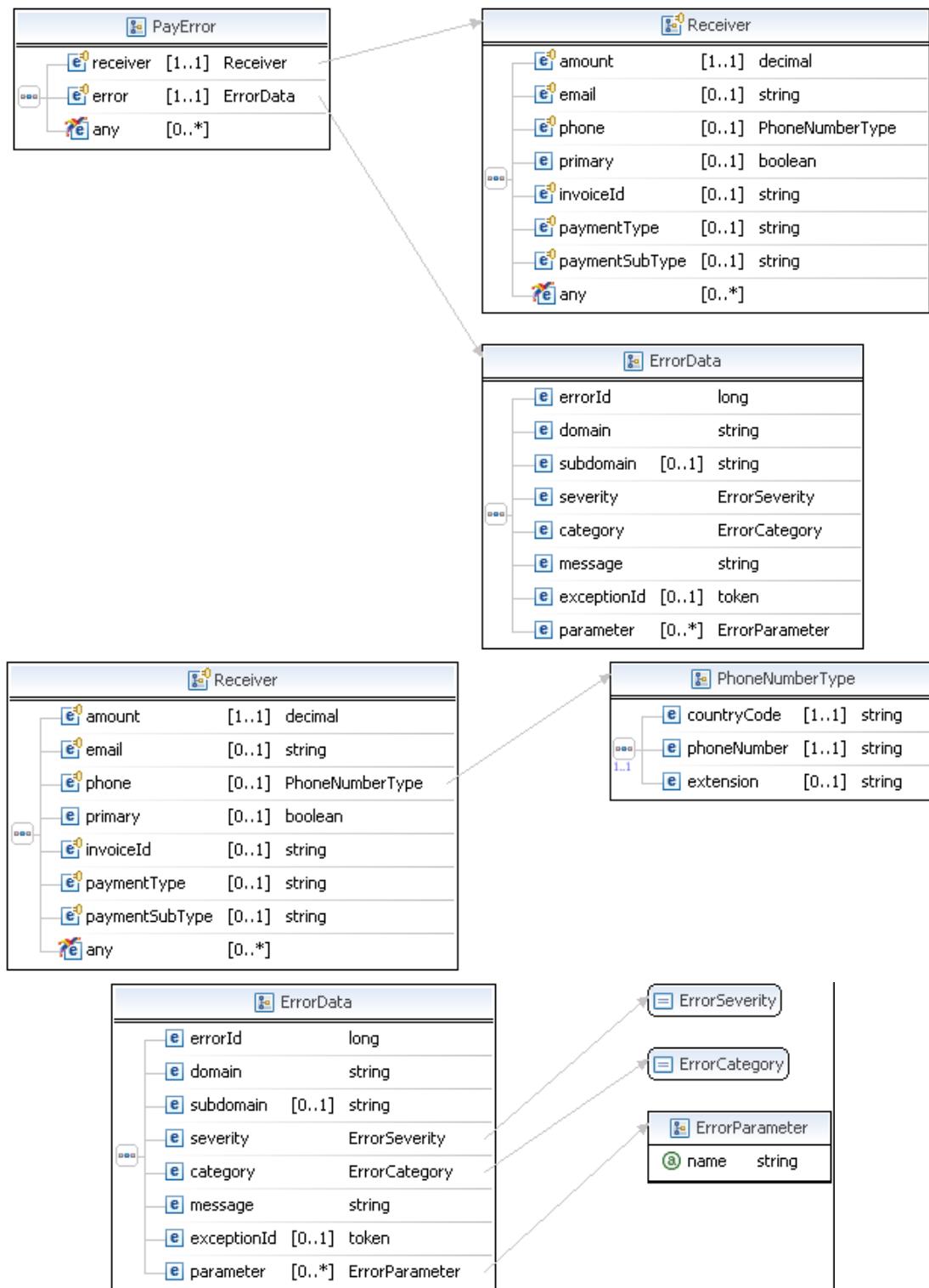
Field	Description
detailLevel	common:DetailLevelCode <i>(Optional)</i> The level of detail required by the client application for component such as Item, Transaction. Possible values are: <ul style="list-style-type: none">• ReturnAll – This value provides the maximum level of detail (default)
errorLanguage	xs:string <i>(Required)</i> The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported

ExecutePaymentResponse Message



ExecutePayment API Operation

ExecutePaymentResponse Message



ExecutePayment API Operation

ExecutePaymentResponse Message

ExecutePaymentResponse Fields

Field	Description
payErrorList	ap:PayErrorList Information about why a payment failed.
paymentExecStatus	xs:string The status of the payment. Possible values are: <ul style="list-style-type: none">• CREATED – The payment request was received; funds will be transferred once the payment is approved• COMPLETED – The payment was successful• INCOMPLETE – Some transfers succeeded and some failed for a parallel payment• ERROR – The payment failed and all attempted transfers failed or all completed transfers were successfully reversed• REVERSALERROR – One or more transfers failed when attempting to reverse a payment <p>IMPORTANT: You must test the value of paymentExecStatus for an error even if responseEnvelope.ack is Success.</p>
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

PayErrorList Fields

Field	Description
payError	PayError indicates the error, if any, that resulted on an attempted payment to a receiver.

PayError Fields

Field	Description
error	Detailed error information.
receiver	ap:Receiver Receiver is the party where funds are transferred to. A primary receiver receives a payment directly from the sender in a chained split payment. A primary receiver should not be specified when making a single or parallel split payment.

ErrorData Fields

Field	Description
category	<p>common:ErrorCategory</p> <p>The location where the error occurred.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • System – The system encountered errors; try again • Application – The application encountered errors; try again • Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	<p>common:ErrorParameter</p> <p>Represents contextual information about the error.</p>
severity	<p>common:ErrorSeverity</p> <p>The severity of the error encountered.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Error – Processing of the request was interrupted • Warning – Processing of the request was completed
subdomain	This field is not used.

Receiver Fields

Field	Description
amount	<p>xs:decimal</p> <p>Amount to be paid to the receiver.</p>
email	<p>xs:string</p> <p>Receiver's email address.</p> <p>Maximum length: 127 characters</p>
invoiceId	<p>xs:string</p> <p>The invoice number for the payment. This data in this field shows on the Transaction Details report.</p> <p>Maximum length: 127 characters</p>

ExecutePayment API Operation

PPFault Message

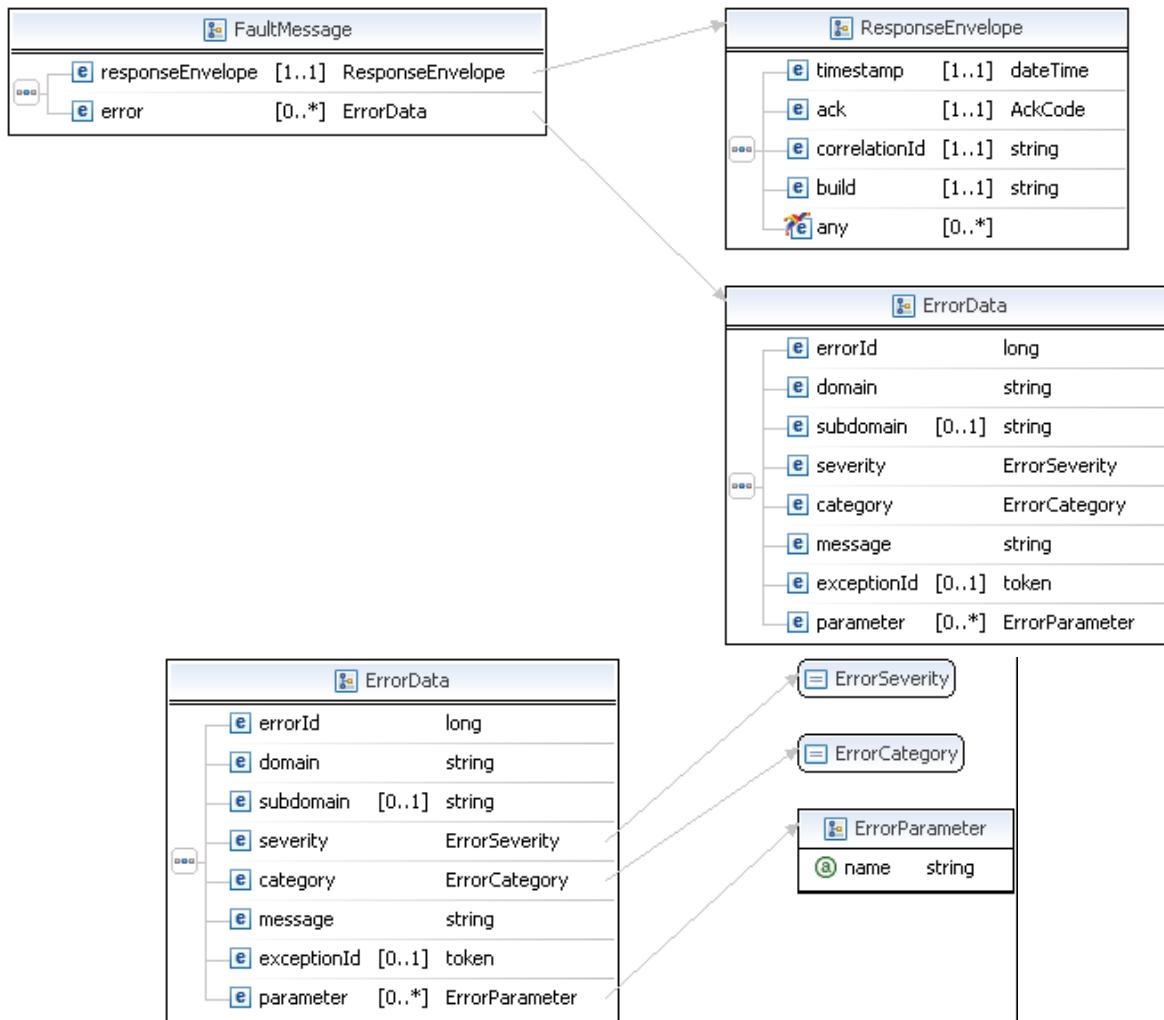
Field	Description
paymentType	<p>xs:string</p> <p>The transaction type for the payment.</p> <p>Possible values are:</p> <ul style="list-style-type: none">• GOODS – This is a payment for non-digital goods• SERVICE – This is a payment for services (default)• PERSONAL – This is a person-to-person payment• CASHADVANCE – This is a person-to-person payment for a cash advance• DIGITALGOODS – This is a payment for digital goods
paymentSubType	<p>xs:string</p> <p>The transaction subtype for the payment.</p>
phone	<p>common:PhoneNumberType</p> <p>The receiver's phone number.</p>
primary	<p>xs:boolean</p> <p>Whether this receiver is the primary receiver. If this field is set to true, this is a chained payment. If this field shows false, this is a simple or parallel payment.</p> <p>Possible values are:</p> <ul style="list-style-type: none">• true – Primary receiver (chained payment)• false – Secondary receiver (simple/parallel payment)

ResponseEnvelope Fields

Field	Description
ack	<p>common:AckCode</p> <p>Acknowledgement code. Possible values are:</p> <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationID	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.



FaultMessage Fields

Field	Description
<code>error</code>	<code>common:ErrorData</code> Detailed error information.
<code>responseEnvelope</code>	<code>common:ResponseEnvelope</code> Common response information, including a timestamp and the response acknowledgement status.

ExecutePayment API Operation

PPFault Message

ErrorData Fields

Field	Description
category	<code>common:ErrorCategory</code> The location where the error occurred. Possible values are: <ul style="list-style-type: none">• System – The system encountered errors; try again• Application – The application encountered errors; try again• Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	<code>common:ErrorParameter</code> Represents contextual information about the error.
severity	<code>common:ErrorSeverity</code> The severity of the error encountered. Possible values are: <ul style="list-style-type: none">• Error – Processing of the request was interrupted• Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	<code>common:AckCode</code> Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

ExecutePayment Errors

Code	Message	Additional Information
500000	There is a system error	
520002	Internal error	
520003	User name/password is incorrect	
520003	Authentication failed. API credentials are incorrect.	Since 1.6.0
520005	Merchant account is locked	
520006	This call is not defined in the database	
529038	There was an error while making this payment	
539012	The preapproval key has not been authorized yet	
539041	The email account is based in a country that is not enabled to receive payments	
539043	The email account is based in a country that is not enabled to send payments	
540031	You do not have permission to execute this payment	
540031	You don't have permission to cancel this preapproval	
540031	You do not have permission to execute this payment Operation is not permitted because the credentials do not match those of the initial operation	Since 1.6.0
550001	User is not allowed to perform this action	
550001	This payment request must be authorized by the sender	
550001	You do not have permission to execute this payment implicitly	Since 1.6.0
559044	Account setting on the receiver prohibited the payment	
560027	The argument value is unsupported	
569000	Split payments are not supported at this time	
569013	The preapproval key has been canceled	
569016	Preapproval PIN functionality is not enabled	
569017	The preapproval key has been suspended	
569018	Preapproved payments have been disabled	
569019	The preapproval has been suspended due to too many PIN failures	
569042	The email account is not confirmed by PayPal	

ExecutePayment API Operation

ExecutePayment Errors

Code	Message	Additional Information
579007	The maximum number of receivers is <number>	
579010	If a preapproval key is specified, the sender's email address must be, too	
579014	The preapproval key specifies a different sender than the payment request	
579017	The amount for the primary receiver must be greater than or equal to the total of other chained receiver amounts	
579024	The preapproval key cannot be used before the start date or after the end date	
579025	The preapproval key cannot be used on this weekday	
579026	The preapproval key cannot be used on this day of the month	
579027	The preapproval key specifies a different currency than the payment request	
579028	The payment amount exceeds the maximum amount per payment	
579030	The number of payments made this period exceeds the maximum number of payments per period	
579031	The total amount of all payments exceeds the maximum total amount for all payments	
579033	The sender and each receiver must have different accounts	
579040	The receivers cannot belong to the same PayPal account	
579042	The tracking ID already exists and cannot be duplicated	
579045	The email account exceeds the receiving limit	
579047	The email account exceeds the purse limit	
579048	The email account exceeds the sending limit	
580001	Invalid request	
580022	Your request was invalid. Check the parameter of the error to see which is invalid	
580022	<ul style="list-style-type: none">• Invalid request parameter: payKey with value <value>• payKey <value> has already been used to make a payment	
580022	The funding plan data is not present	Since 1.6.0. Tried to pass in a non-zero funding plan and there are no funding plans present
580023	Invalid request	
580027	The argument is unsupported	
580028	The URL <value> is malformed	

Code	Message	Additional Information
580029	Invalid request	
580029	A pay key, transaction ID, or tracking ID is missing	Since 1.6.0
589009	This payment cannot be processed because no payment source is available	
589023	If a fractional amount is rounded due to currency conversion, funds could be lost	
589039	The email address is invalid. It may not be registered in PayPal's system yet	

6

GetPaymentOptions API Operation

You use the GetPaymentOptions API operation to retrieve the payment options passed with the SetPaymentOptionsRequest.

GetPaymentOptionsRequest Message



GetPaymentOptionsRequest Fields

Field	Description
payKey	xs:string <i>(Required)</i> The pay key that identifies the payment for which you want to get payment options. This is the pay key you used to set the payment options.
requestEnvelope	common:requestEnvelope <i>(Required)</i> Information common to each API operation, such as the language in which an error message is returned.

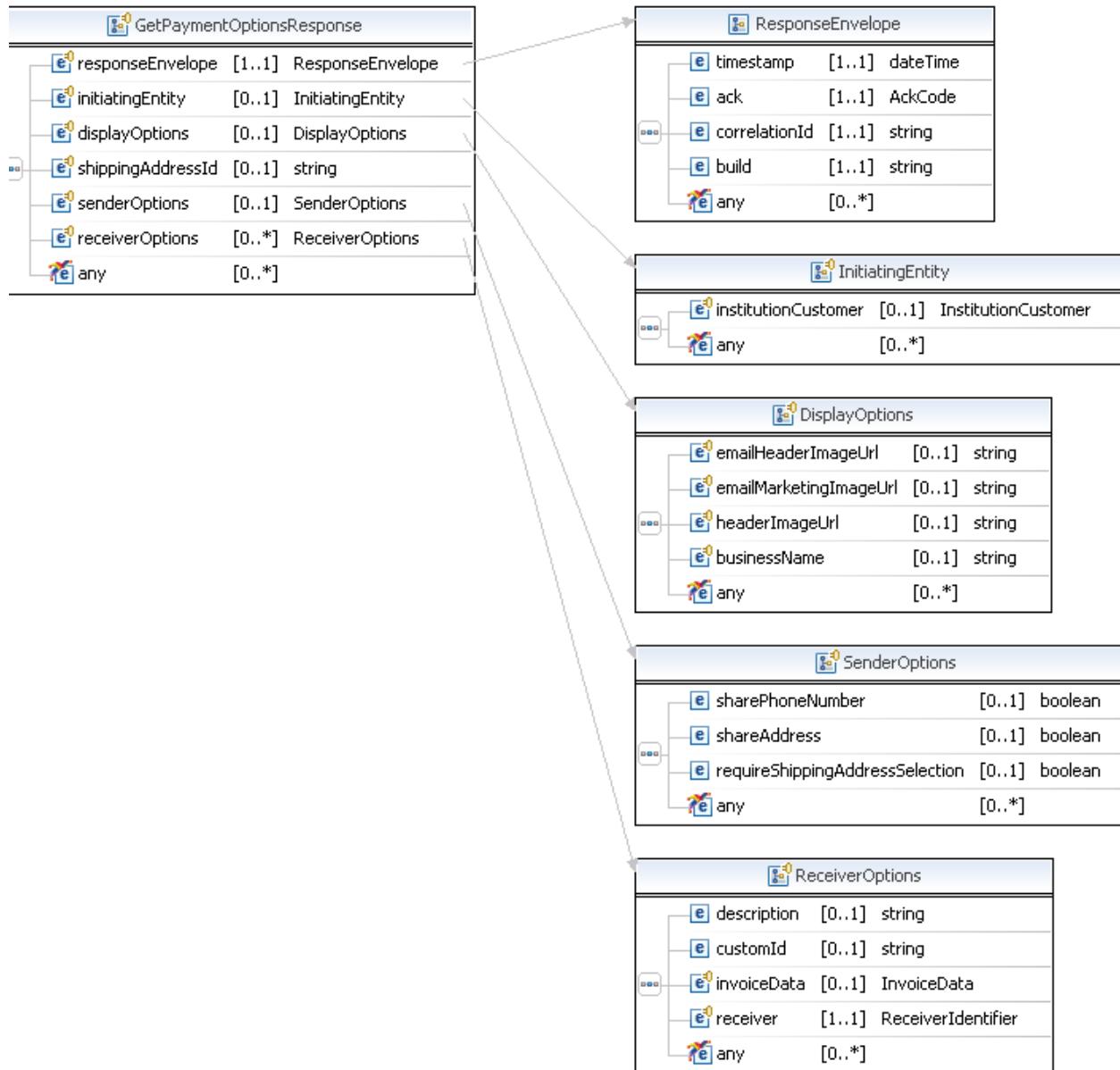
RequestEnvelope Fields

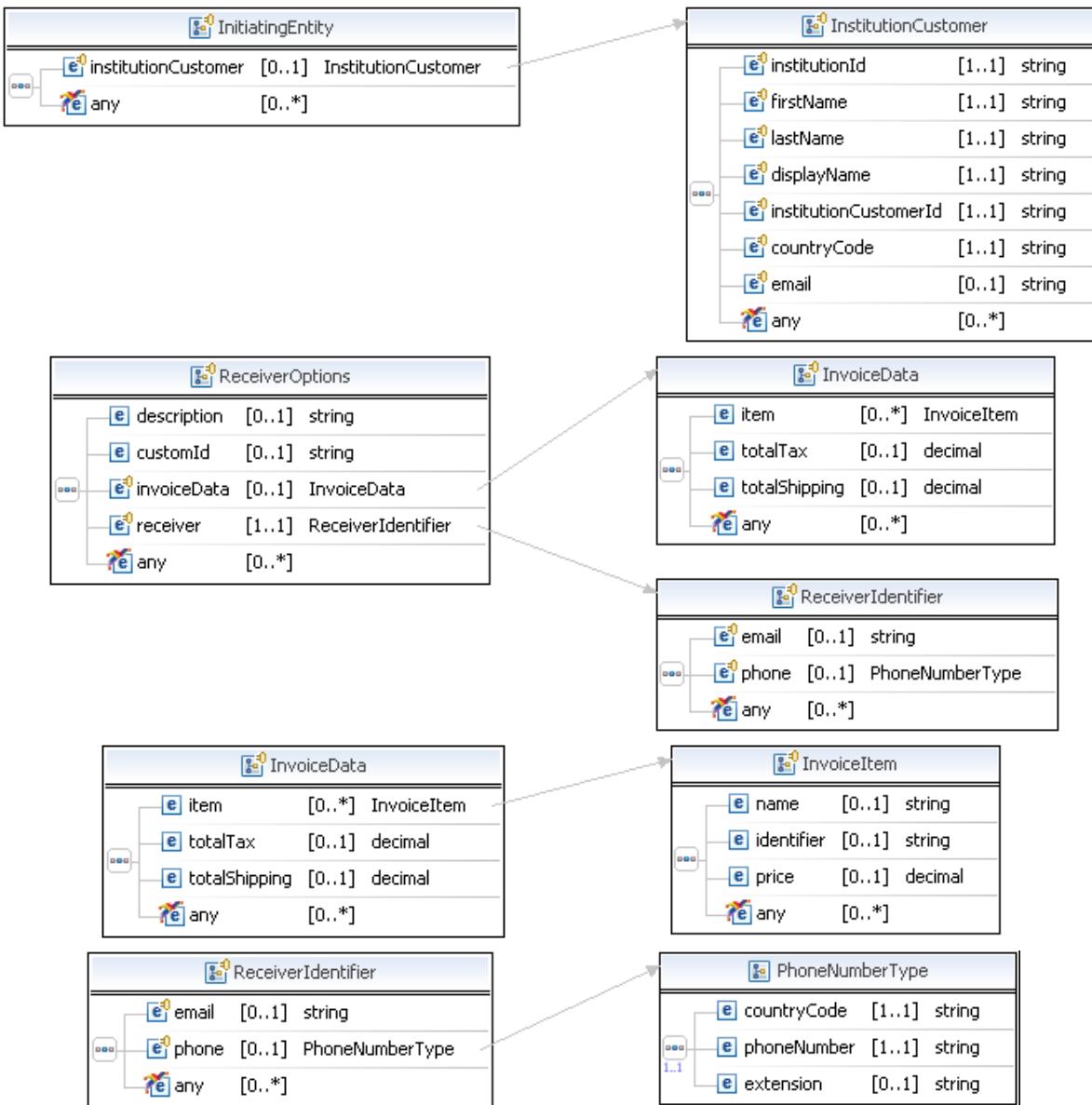
Field	Description
detailLevel	common:DetailLevelCode <i>(Optional)</i> The level of detail required by the client application for component such as Item, Transaction. Possible values are: <ul style="list-style-type: none">ReturnAll – This value provides the maximum level of detail (default)
errorLanguage	xs:string <i>(Required)</i> The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported

GetPaymentOptions API Operation

GetPaymentOptionsResponse Message

GetPaymentOptionsResponse Message





GetPaymentOptionsResponse Fields

Field	Description
payKey	xs:string The pay key that identifies the payment for which you set payment options.
displayOptions	ap:DisplayOptions Specifies display items in payment flows and emails.

GetPaymentOptions API Operation

GetPaymentOptionsResponse Message

Field	Description
initiatingEntity	ap:InitiatingEntity The PayPal financial partner that is initiating the payment. Financial partners must first be set up in the Admin tool before they can use the PayPal APIs.
shippingAddressId	xs:string Sender's shipping address ID.
senderOptions	ap:SenderOptions Specifies information to be provided by the sender.
receiverOptions	ap:ReceiverOptions Specifies information about each receiver.
requestEnvelope	common:requestEnvelope Information common to each API operation, such as the language in which an error message is returned.

DisplayOptions Fields

Field	Description
emailHeaderImageUrl	xs:string The URL of the image that displays in the header of customer emails. The image dimensions are 43 pixels high x 240 pixels wide.
emailMarketingImageUrl	xs:string The URL of the image that displays in customer emails. The image dimensions are 80 pixels high x 530 pixels wide.
emailHeaderImageUrl	xs:string The URL of the image that displays in the header of a payment page. If set, it overrides the header image URL specified in your account's Profile. The image dimensions are 750 pixels high x 90 pixels wide.
businessName	xs:string The business name to display.

InitiatingEntity Fields

Field	Description
institutionCustomer	ap:InstitutionCustomer <i>(Optional)</i> Details about the party that initiated this payment. This payment is made by the API caller on behalf of the initiating party. The initiating party can be an institution or a customer of the institution. The initiating party must be set up by PayPal Merchant Services.

Institution Customer Fields

Field	Description
countryCode	xs:string <i>(Required)</i> The two-character country code of the home country of the end consumer
displayName	xs:string <i>(Required)</i> The full name of the consumer as known by the institution. Maximum length: 200 characters
email	xs:string <i>(Optional)</i> The email address of the consumer as known by the institution. Maximum length: 127 characters
firstName	xs:string <i>(Required)</i> The first name of the consumer as known by the institution. Maximum length: 64 characters
institutionCustomerId	xs:string <i>(Required)</i> The unique identifier assigned to the consumer by the institution. Maximum length: 64 characters
institutionId	xs:string <i>(Required)</i> The unique identifier assigned to the institution. Maximum length: 64 characters
lastName	xs:string <i>(Required)</i> The last name of the consumer as known by the institution. Maximum length: 64 characters

SenderOptions Fields

Field	Description
sharePhoneNumber	xs:boolean If true, share the phone number with receivers; default is false.
shareAddress	xs:boolean If true, share the address with receivers; default is false.
requireShippingAddressSelection	xs:boolean If true, require the sender to select a shipping address during the payment flow; default is false.

GetPaymentOptions API Operation

GetPaymentOptionsResponse Message

ReceiverOptions Fields

Field	Description
description	xs:string A description you want to associate with the payment.
customId	xs:string An external reference or identifier you want to associate with the payment.
invoiceData	ap:InvoiceData Item information for the payment, which could appear on an invoice.
receiver	ap:ReceiverIdentifier Receiver ID associated with these options.

InvoiceData Fields

Field	Description
item	ap:InvoiceItem Any number of items associated with the payment.
totalTax	xs:decimal Total tax associated with the payment.
totalShipping	xs:decimal Total shipping charge associated with the payment.

InvoiceItem Fields

Field	Description
name	xs:string Name of item.
identifier	xs:string External reference to item or item ID.
price	xs:decimal Price of item.

ReceiverIdentifier Fields

Field	Description
email	<p>xs:string</p> <p>Receiver's email address.</p> <p>Maximum length: 127 characters</p>
phone	<p>common:PhoneNumberType</p> <p>Receiver's phone number.</p>

PhoneNumberType Response Fields

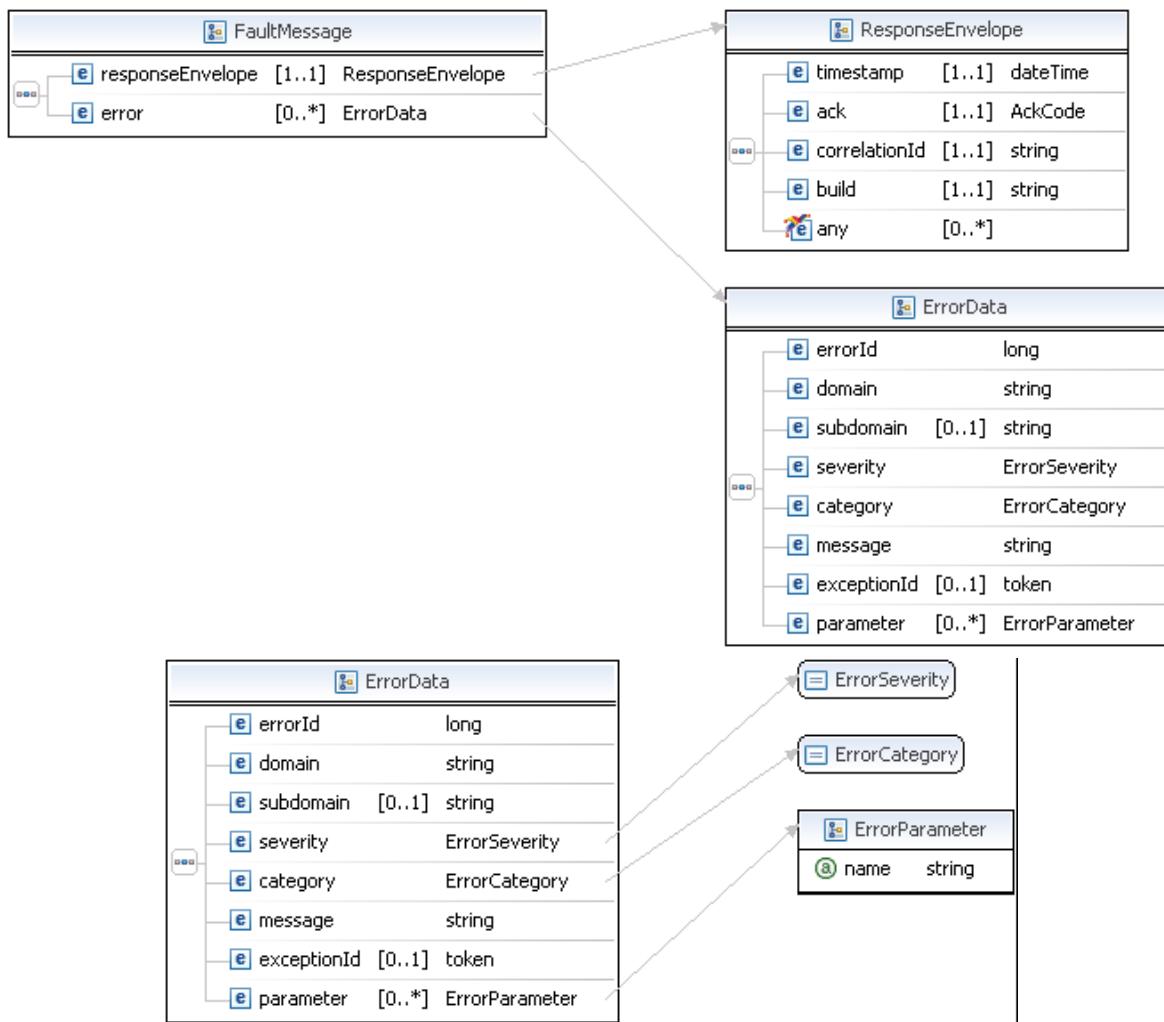
Field	Description
countryCode	<p>xs:string</p> <p>Telephone country code.</p>
phoneNumber	<p>xs:string</p> <p>Telephone number.</p>
extension	<p>xs:string</p> <p>Telephone extension.</p>

ResponseEnvelope Fields

Field	Description
ack	<p>common:AckCode</p> <p>Acknowledgement code. Possible values are:</p> <ul style="list-style-type: none"> • Success – Operation completed successfully • Failure – Operation failed • Warning – Warning • SuccessWithWarning – Operation completed successfully; however, there is a warning message • FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.



FaultMessage Fields

Field	Description
<code>error</code>	<code>common:ErrorData</code> Detailed error information.
<code>responseEnvelope</code>	<code>common:ResponseEnvelope</code> Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	<p>common:ErrorCategory</p> <p>The location where the error occurred.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • System – The system encountered errors; try again • Application – The application encountered errors; try again • Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	<p>common:ErrorParameter</p> <p>Represents contextual information about the error.</p>
severity	<p>common:ErrorSeverity</p> <p>The severity of the error encountered.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Error – Processing of the request was interrupted • Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	<p>common:AckCode</p> <p>Acknowledgement code. Possible values are:</p> <ul style="list-style-type: none"> • Success – Operation completed successfully • Failure – Operation failed • Warning – Warning • SuccessWithWarning – Operation completed successfully; however, there is a warning message • FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

GetPaymentOptions Errors

Code	Message	Additional Information
520002	Internal error	
520003	Authentication failed. API credentials are incorrect.	Since 1.6.0
540031	You do not have permission to get these payment options	
540031	You do not have permission to get these payment options Operation is not permitted because the credentials do not match those of the initial operation	Since 1.6.0
550001	User is not allowed to perform this action	
580001	Invalid request: <value>	Since 1.6.0
580022	Invalid request parameter: payKey with value <value>	

7

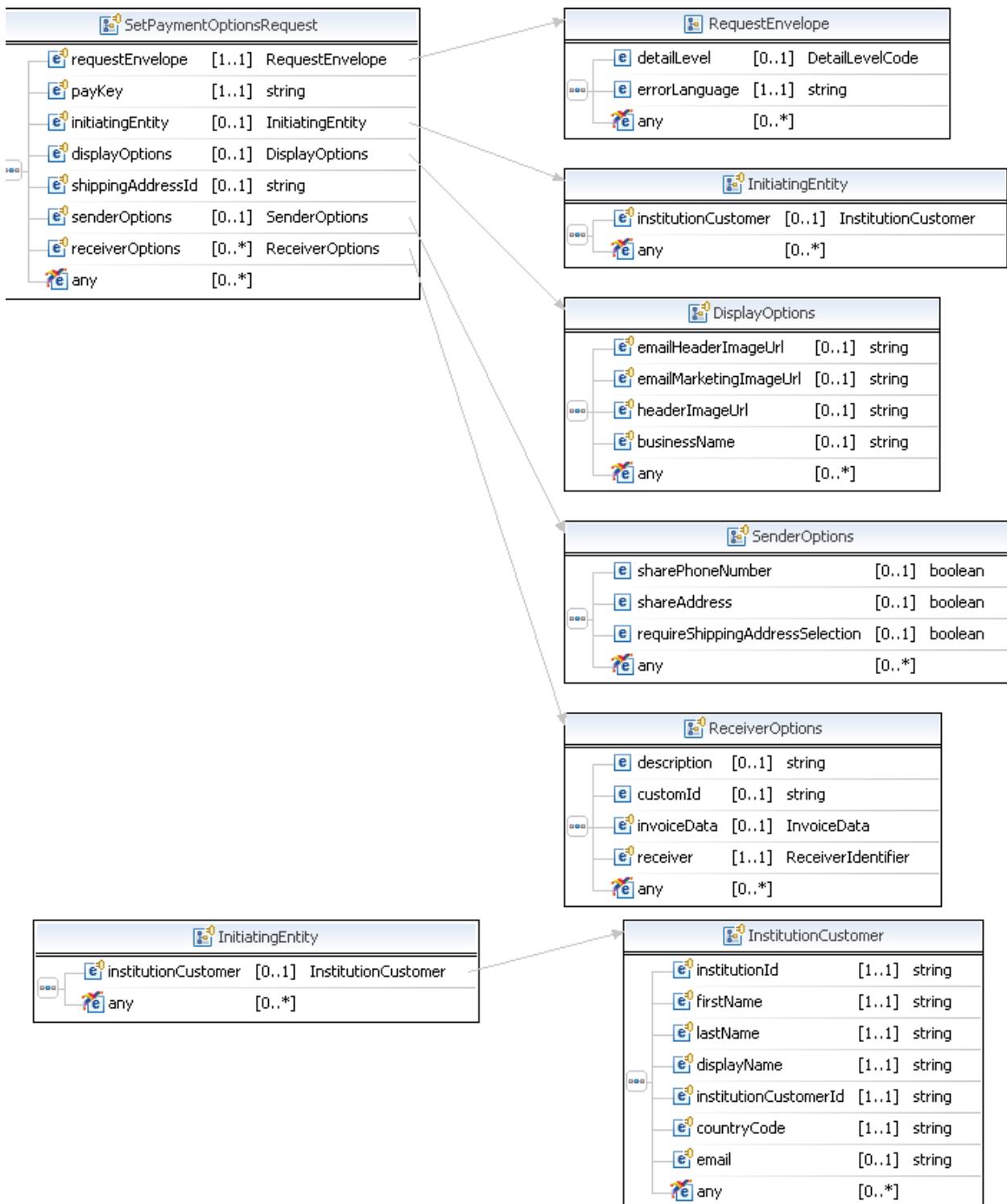
SetPaymentOptions API Operation

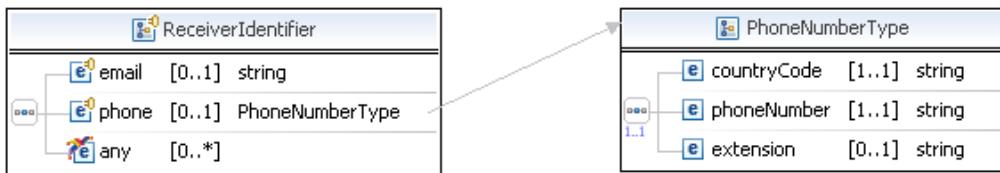
You use the SetPaymentOptions API operation to specify settings for a payment of the `actionType CREATE`. This `actionType` is specified in the PayRequest message.

SetPaymentsOptionsRequest Message

SetPaymentOptions API Operation

SetPaymentsOptionsRequest Message





SetPaymentOptionsRequest Fields

Field	Description
payKey	xs:string <i>(Required)</i> The pay key that identifies the payment for which you want to set payment options. This is the pay key returned in the PayResponse message.
displayOptions	ap:DisplayOptions <i>(Optional)</i> Specifies display items in payment flows and emails.
initiatingEntity	ap:InitiatingEntity <i>(Optional)</i> The PayPal financial partner that is initiating the payment. Financial partners must first be set up in the Admin tool before they can use the PayPal APIs.
shippingAddressId	xs:string <i>(Optional)</i> Sender's shipping address ID.
senderOptions	ap:SenderOptions <i>(Optional)</i> Specifies information to be provided by the sender.
receiverOptions	ap:ReceiverOptions <i>(Optional)</i> Specifies information about each receiver.
requestEnvelope	common:requestEnvelope <i>(Required)</i> Information common to each API operation, such as the language in which an error message is returned.

DisplayOptions Fields

Field	Description
emailHeaderImageUrl	xs:string <i>(Optional)</i> The URL of the image that displays in the header of customer emails. The image dimensions are 43 pixels high x 240 pixels wide.
emailMarketingImageUrl	xs:string <i>(Optional)</i> The URL of the image that displays in customer emails. The image dimensions are 80 pixels high x 530 pixels wide.

SetPaymentOptions API Operation

SetPaymentsOptionsRequest Message

Field	Description
headerImageUrl	<code>xs:string</code> <i>(Optional)</i> The URL of the image that displays in the header of a payment page. If set, it overrides the header image URL specified in your account's Profile. The image dimensions are 750 pixels high x 90 pixels wide.
businessName	<code>xs:string</code> <i>(Optional)</i> The business name to display.

InitiatingEntity Fields

Field	Description
institutionCustomer	<code>ap:InstitutionCustomer</code> <i>(Optional)</i> Details about the party that initiated this payment. This payment is made by the API caller on behalf of the initiating party. The initiating party can be an institution or a customer of the institution. The initiating party must be set up by PayPal Merchant Services.

Institution Customer Fields

Field	Description
countryCode	<code>xs:string</code> <i>(Required)</i> The two-character country code of the home country of the end consumer
displayName	<code>xs:string</code> <i>(Required)</i> The full name of the consumer as known by the institution. Maximum length: 200 characters
email	<code>xs:string</code> <i>(Optional)</i> The email address of the consumer as known by the institution. Maximum length: 127 characters
firstName	<code>xs:string</code> <i>(Required)</i> The first name of the consumer as known by the institution. Maximum length: 64 characters
institutionCustomerId	<code>xs:string</code> <i>(Required)</i> The unique identifier assigned to the consumer by the institution. Maximum length: 64 characters
institutionId	<code>xs:string</code> <i>(Required)</i> The unique identifier assigned to the institution. Maximum length: 64 characters

Field	Description
lastName	<p>xs:string</p> <p>(Required) The last name of the consumer as known by the institution.</p> <p>Maximum length: 64 characters</p>

SenderOptions Fields

Field	Description
sharePhoneNumber	<p>xs:boolean</p> <p>(Optional) If true, share the phone number with receivers; default is false.</p>
shareAddress	<p>xs:boolean</p> <p>(Optional) If true, share the address with receivers; default is false.</p>
requireShippingAddressSelection	<p>xs:boolean</p> <p>(Optional) If true, require the sender to select a shipping address during the payment flow; default is false.</p>

ReceiverOptions Fields

Field	Description
description	<p>xs:string</p> <p>(Optional) A description you want to associate with the payment.</p>
customId	<p>xs:string</p> <p>(Optional) An external reference or identifier you want to associate with the payment.</p>
invoiceData	<p>ap:InvoiceData</p> <p>(Optional) Item information for the payment, which could appear on an invoice.</p>
receiver	<p>ap:ReceiverIdentifier</p> <p>(Optional) Receiver ID associated with these options.</p>

InvoiceData Fields

Field	Description
item	<p>ap:InvoiceItem</p> <p>(Optional) Any number of items associated with the payment.</p>

SetPaymentOptions API Operation

SetPaymentsOptionsRequest Message

Field	Description
totalTax	xs:decimal <i>(Optional)</i> Total tax associated with the payment.
totalShipping	xs:decimal <i>(Optional)</i> Total shipping charge associated with the payment.

InvoicelItem Fields

Field	Description
name	xs:string <i>(Optional)</i> Name of item.
identifier	xs:string <i>(Optional)</i> External reference to item or item ID.
price	xs:decimal <i>(Optional)</i> Price of item.

ReceiverIdentifier Fields

Field	Description
email	xs:string <i>(Optional)</i> Receiver's email address. Maximum length: 127 characters
phone	common:PhoneNumberType <i>(Optional)</i> Receiver's phone number.

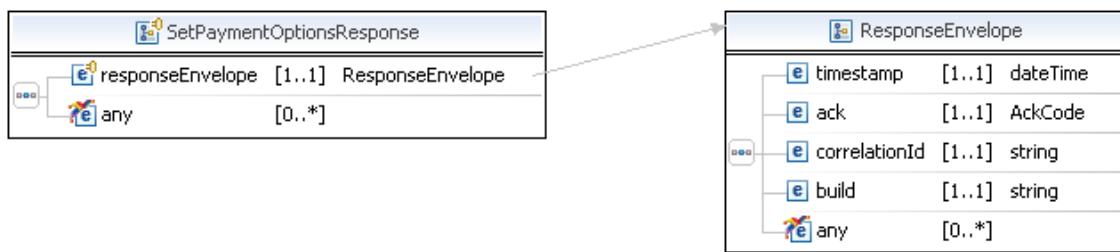
PhoneNumberType Fields

Field	Description
countryCode	xs:string <i>(Required)</i> Telephone country code.
phoneNumber	xs:string <i>(Required)</i> Telephone number.
extension	xs:string <i>(Optional)</i> Telephone extension.

RequestEnvelope Fields

Field	Description
detailLevel	<p>common:DetailLevelCode</p> <p>(Optional) The level of detail required by the client application for component such as Item, Transaction. Possible values are:</p> <ul style="list-style-type: none"> • ReturnAll – This value provides the maximum level of detail (default)
errorLanguage	<p>xs:string</p> <p>(Required) The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported</p>

SetPaymentOptionsResponse Message



ResponseEnvelope Fields

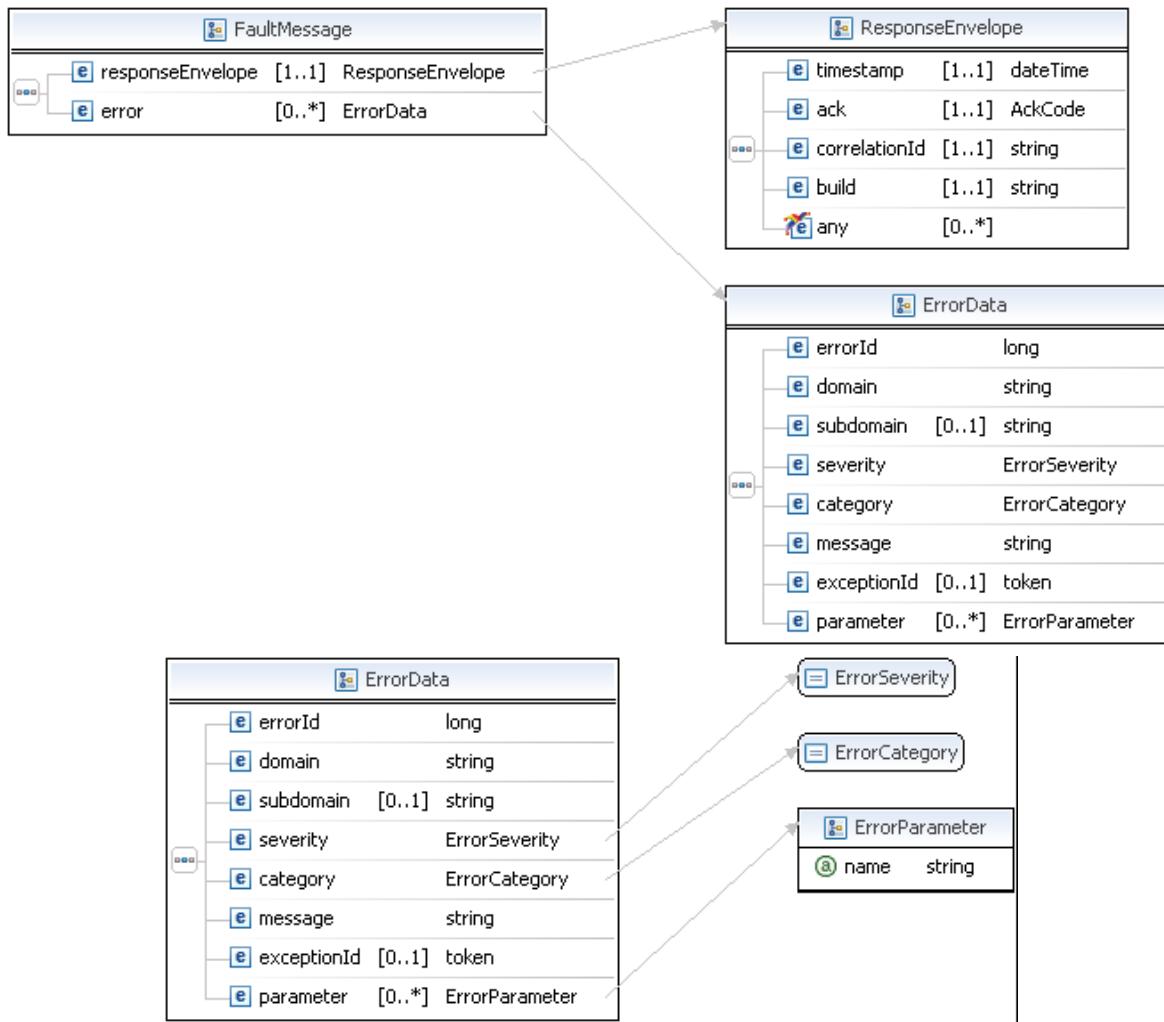
Field	Description
ack	<p>common:AckCode</p> <p>Acknowledgement code. Possible values are:</p> <ul style="list-style-type: none"> • Success – Operation completed successfully • Failure – Operation failed • Warning – Warning • SuccessWithWarning – Operation completed successfully; however, there is a warning message • FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

SetPaymentOptions API Operation

PPFault Message

PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.



FaultMessage Fields

Field	Description
error	common:ErrorData Detailed error information.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	<p>common:ErrorCategory</p> <p>The location where the error occurred.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • System – The system encountered errors; try again • Application – The application encountered errors; try again • Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	<p>common:ErrorParameter</p> <p>Represents contextual information about the error.</p>
severity	<p>common:ErrorSeverity</p> <p>The severity of the error encountered.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Error – Processing of the request was interrupted • Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	<p>common:AckCode</p> <p>Acknowledgement code. Possible values are:</p> <ul style="list-style-type: none"> • Success – Operation completed successfully • Failure – Operation failed • Warning – Warning • SuccessWithWarning – Operation completed successfully; however, there is a warning message • FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

SetPaymentOptions Errors

Code	Message	Additional Information
520002	Internal error	
520003	Authentication failed. API credentials are incorrect.	Since 1.6.0
550001	User is not allowed to perform this action	
550001	This payment request must be authorized by the sender	
540031	You do not have permission to set these payment options	
540031	You do not have permission to set these payment options Operation is not permitted because the credentials do not match those of the initial operation	Since 1.6.0
580001	Invalid request: <value>	Since 1.6.0
580022	<ul style="list-style-type: none"> • Invalid request parameter: payKey with value <value> • payKey <value> has already been used to make a payment • displayOptions has already been set and cannot be set multiple times • initiationEntity has already been set and cannot be set multiple times • Invalid Request Parameter: institutionId with value <value> • The institution ID <value> is not activated 	
580022	<ul style="list-style-type: none"> • Invalid request parameter: <value> • Invalid Request Parameter: institutionId with value <value> • The institution id <value> is not activated • The clientDetails parameter is missing • Invalid request parameter: payKey with value <value> • Invalid Phone Country Code <value> • Invalid Country Code <value> for Receiver • Invalid Receiver Phone Type <value> <value> <value> • Invalid Phone Format <value> <value> • Invalid Phone Number <value> <value> • Invalid Receiver Phone Number <value> <value> • Invalid Area or Exchange Code <value> <value> • <value> has already been set and cannot be set multiple times • Invalid request parameter: <value> with value <value> • Invalid request parameter: <value> <value> is invalid • Invalid request parameter: <value> <value> is duplicated • Invalid request parameter: <value> <value> cannot be negative or zero 	Since 1.6.0

Code	Message	Additional Information
580027	<ul style="list-style-type: none">• Currently the system does not accept the currency <value>• Unsupported country code <value>• Currently the system does not accept the currency <value>• The parameter is not supported	Since 1.6.0
580028	The URL <value> is malformed	

8

Preapproval API Operation

Use the Preapproval API operation to set up an agreement between yourself and a sender for making payments on the sender's behalf.

You set up a preapprovals for a specific maximum amount over a specific period of time and, optionally, by any of the following constraints: the number of payments, a maximum per-payment amount, a specific day of the week or the month, and whether or not a PIN is required for each payment request.

Preapproval Overview

To set up a preapproval request, you must specify values for the following fields:

Required preapproval fields

Field	Description
endingDate	Last date for which the preapproval is valid. It cannot be later than one year from the starting date. Contact PayPal if you do not want to specify an ending date.
startingDate	First date for which the preapproval is valid. It cannot be before today's date or after the ending date.
maxTotalAmountOfAllPayments	The preapproved maximum total amount of all payments. It cannot exceed \$2,000 USD or its equivalent in other currencies. Contact PayPal if you do not want to specify a maximum amount.
currencyCode	The code for the currency in which the payment is made; you can specify only one currency, regardless of the number of receivers
cancelUrl	URL to redirect the sender's browser to after canceling the preapproval
returnUrl	URL to redirect the sender's browser to after the sender has logged into PayPal and confirmed the preapproval
requestEnvelope.errorLanguage	The code for the language in which errors are returned, which must be en_US.

Preapproval Notifications

Notifications are sent after preapproval is complete:

Preapproval API Operation

PreapprovalRequest Message

- PayPal sends an email to the sender that confirmed the approval.
- PayPal sends an IPN message to the URL specified in the `ipnNotificationUrl` field of the Preapproval request.

Additional Notes About the PreApproval API Operation

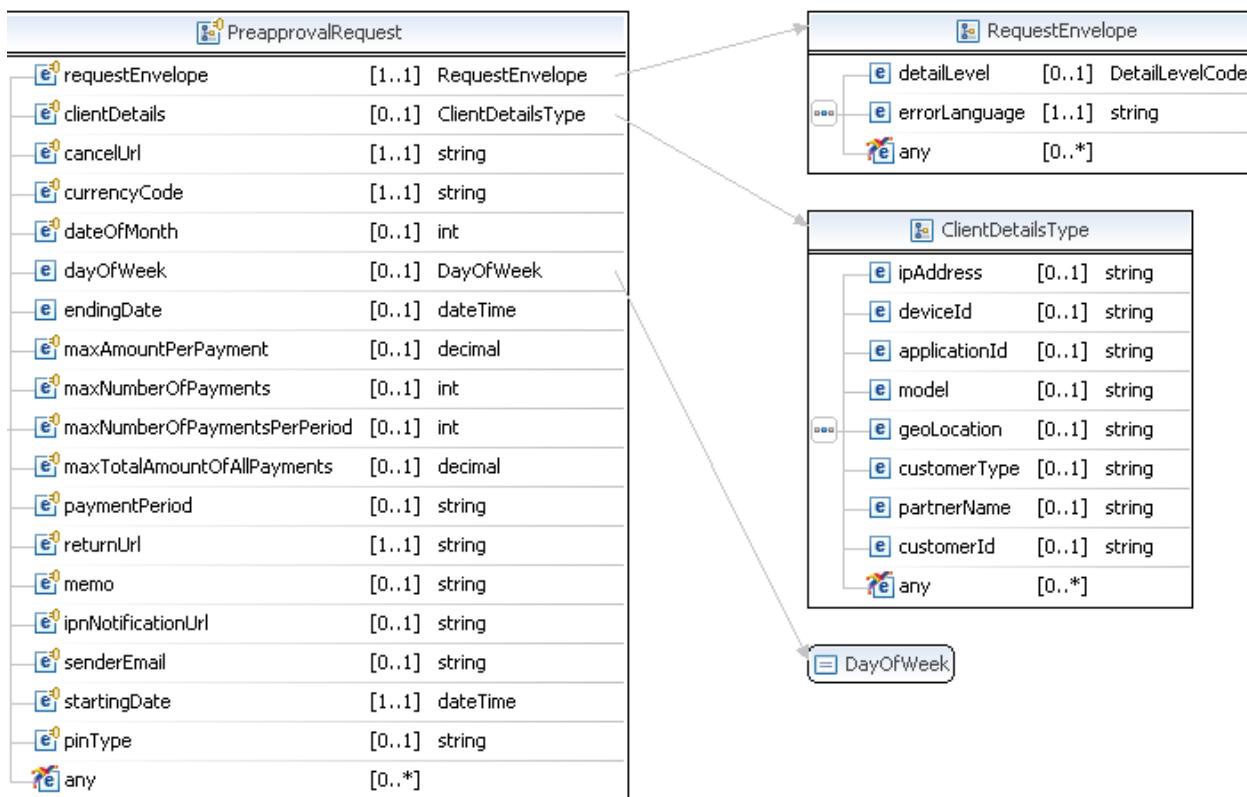
1. Preapproval constraints are additive; thus, for example, if you specify a preapproval that allows payments only on Fridays and on the 13th day of the month, the preapproval would be valid only on Friday the 13th of months within the specified time period.
2. The `startDate` and `endDate` can be in either Zulu or GMT offset formats, as in the following respective examples:

2010-09-10Z

2010-09-10T17:24:03.874-07:00

PreapprovalRequest Message

The `PreapprovalRequest` message contains the fields to set up a preapproval agreement between yourself and a sender.



PreapprovalRequest Fields

Field	Description
<code>cancelUrl</code>	<code>xs:string</code> <i>(Required)</i> The URL to which the sender's browser is redirected after the sender cancels the preapproval at paypal.com. Maximum length: 1024 characters
<code>clientDetails</code>	<code>common:ClientDetailsType</code> <i>(Required)</i> Information about the sender.

Preapproval API Operation

PreapprovalRequest Message

Field	Description
currencyCode	<p>xs:string</p> <p>(Required) The currency code. Allowable values are:</p> <ul style="list-style-type: none">• Australian Dollar – AUD• Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none">• Canadian Dollar – CAD• Czech Koruna – CZK• Danish Krone – DKK• Euro – EUR• Hong Kong Dollar – HKD• Hungarian Forint – HUF• Israeli New Sheqel – ILS• Japanese Yen – JPY• Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none">• Mexican Peso – MXN• Norwegian Krone – NOK• New Zealand Dollar – NZD• Philippine Peso – PHP• Polish Zloty – PLN• Pound Sterling – GBP• Singapore Dollar – SGD• Swedish Krona – SEK• Swiss Franc – CHF• Taiwan New Dollar – TWD• Thai Baht – THB• U.S. Dollar – USD
dateOfMonth	<p>xs:int</p> <p>(Optional) The day of the month on which a monthly payment is to be made. Allowable values are numbers between 0 and 31. A number between 1 and 31 indicates the date of the month. Specifying 0 indicates that payment can be made on any day of the month.</p>

Field	Description
dayOfWeek	<p>common:DayOfWeek</p> <p>(Optional) The day of the week that a weekly payment is to be made.</p> <p>Allowable values are:</p> <ul style="list-style-type: none"> ● NO_DAY_SPECIFIED ● SUNDAY ● MONDAY ● TUESDAY ● WEDNESDAY ● THURSDAY ● FRIDAY ● SATURDAY
endingDate	<p>xs:dateTime</p> <p>(Optional) Last date for which the preapproval is valid. It cannot be later than one year from the starting date. You must specify a value unless you have specific permission from PayPal.</p>
ipnNotificationUrl	<p>xs:string</p> <p>(Optional) The URL to which you want all IPN messages for this preapproval to be sent. This URL supersedes the IPN notification URL in your profile.</p> <p>Maximum length: 1024 characters</p>
maxAmountPerPayment	<p>xs:decimal</p> <p>(Optional) The preapproved maximum amount per payment. It cannot exceed the preapproved maximum total amount of all payments.</p>
maxNumberOfPayments	<p>xs:int</p> <p>(Optional) The preapproved maximum number of payments. It cannot exceed the preapproved maximum total number of all payments.</p>
maxNumberOfPaymentsPerPeriod	<p>xs:int</p> <p>(Optional) The preapproved maximum number of all payments per period. You must specify a value unless you have specific permission from PayPal.</p>
maxTotalAmountOfAllPayments	<p>xs:decimal</p> <p>(Optional) The preapproved maximum total amount of all payments. It cannot exceed \$2,000 USD or its equivalent in other currencies.</p>
memo	<p>xs:string</p> <p>(Optional) A note about the preapproval.</p> <p>Maximum length: 1000 characters, including newline characters</p>

Preapproval API Operation

PreapprovalRequest Message

Field	Description
paymentPeriod	<p>xs:string</p> <p>(Optional) The payment period. It is one of the following values:</p> <ul style="list-style-type: none">• NO_PERIOD_SPECIFIED• DAILY – Each day• WEEKLY – Each week• BIWEEKLY – Every other week• SEMIMONTHLY – Twice a month• MONTHLY – Each month• ANNUALLY – Each year
pinType	<p>xs:string</p> <p>(Optional) Whether a personal identification number (PIN) is required. It is one of the following values:</p> <ul style="list-style-type: none">• NOT_REQUIRED – A PIN is not required (default)• REQUIRED – A PIN is required; the sender must specify a PIN when setting up the preapproval on PayPal
requestEnvelope	<p>common:RequestEnvelope</p> <p>(Required) Information common to each API operation, such as the language in which an error message is returned.</p>
returnUrl	<p>xs:string</p> <p>(Required) The URL to which the sender's browser is redirected after the sender approves the preapproval on paypal.com.</p> <p>Maximum length: 1024 characters</p>
senderEmail	<p>xs:string</p> <p>(Optional) Sender's email address. If not specified, the email address of the sender who logs in to approve the request becomes the email address associated with the preapproval key.</p> <p>Maximum length: 127 characters</p>
startingDate	<p>xs:dateTime</p> <p>(Required) First date for which the preapproval is valid. It cannot be before today's date or after the ending date.</p>

ClientDetails Fields

Field	Description
applicationId	<p>xs:string</p> <p>(Optional) Your application's identification, such as the name of your application</p>
customerId	<p>xs:string</p> <p>(Optional) Your ID for this sender</p> <p>Maximum length: 127 characters</p>

Field	Description
customerType	xs:string <i>(Optional)</i> Your identification of the type of customer Maximum length: 127 characters
deviceId	xs:string <i>(Optional)</i> Sender's device ID, such as a mobile device's IMEI number or a web browser cookie. If a device ID was passed with the PayRequest, use the same ID here. Maximum length: 127 characters
geoLocation	xs:string <i>(Optional)</i> Sender's geographic location Maximum length: 127 characters
ipAddress	xs:string <i>(Optional)</i> Sender's IP address. If an IP address was passed with the PayRequest, use the same ID here.
model	xs:string <i>(Optional)</i> A sub-identification of the application Maximum length: 127 characters
partnerName	xs:string <i>(Optional)</i> Your organization's name or ID Maximum length: 127 characters

RequestEnvelope Fields

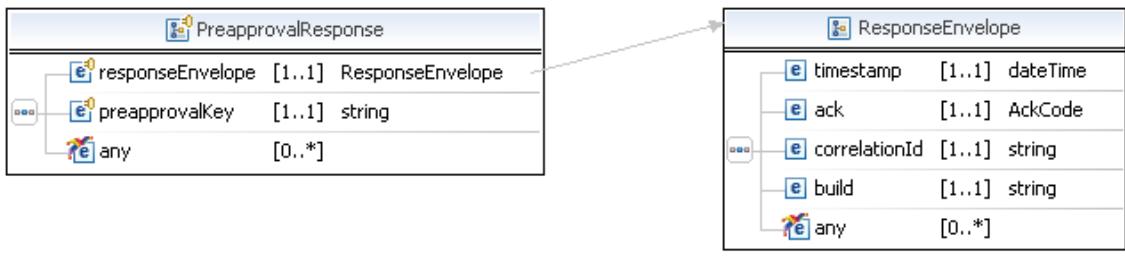
Field	Description
detailLevel	common:DetailLevelCode <i>(Optional)</i> The level of detail required by the client application for component such as Item, Transaction. Possible values are: <ul style="list-style-type: none">● ReturnAll – This value provides the maximum level of detail (default)
errorLanguage	xs:string <i>(Required)</i> The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported

PreapprovalResponse Message

The PreapprovalResponse message contains a preapproval key. This key uniquely identifies the preapproval for this request, and can be used in other Adaptive Payment APIs to identify this specific preapproval.

Preapproval API Operation

PPFault Message



PreapprovalResponse Fields

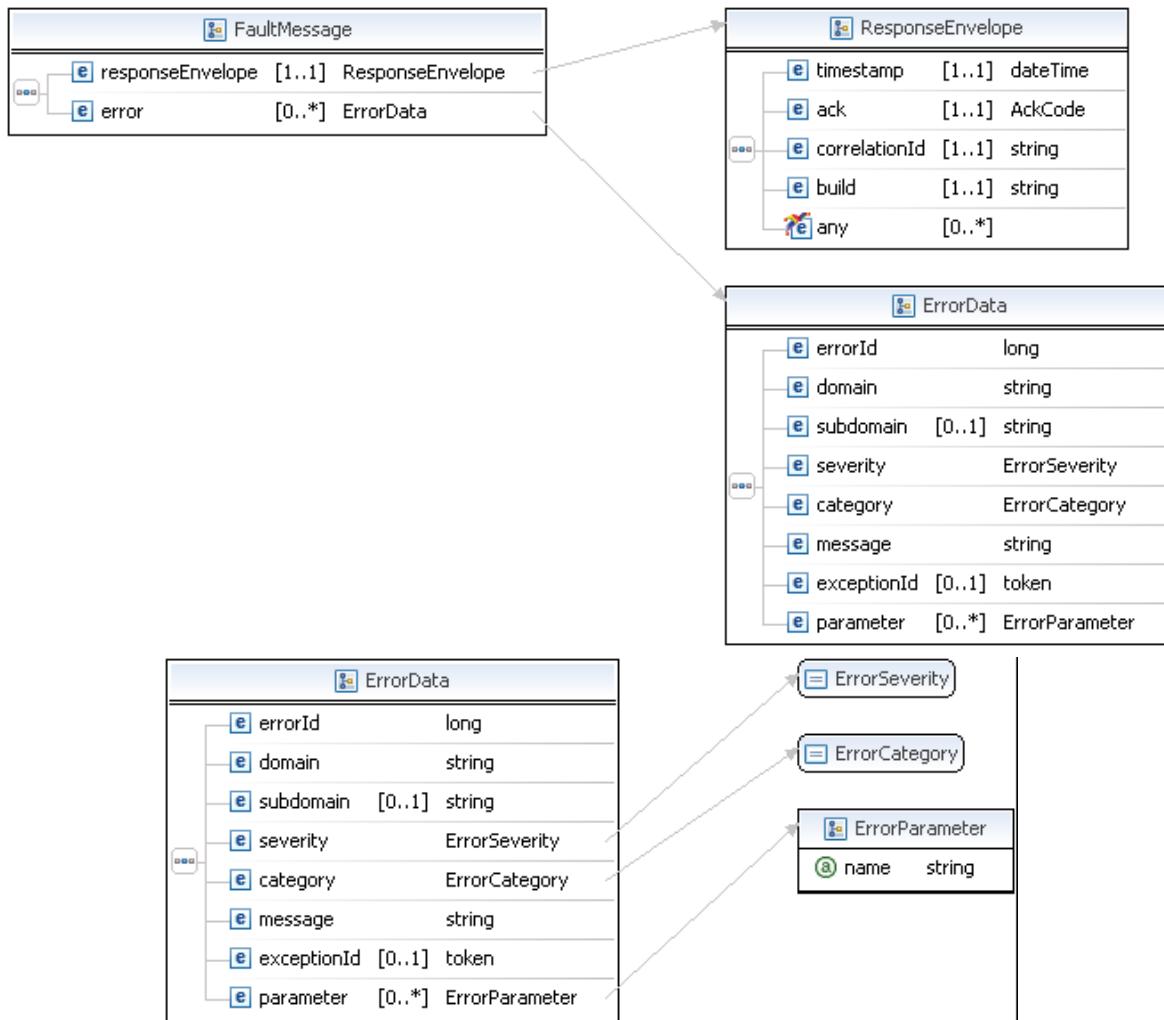
Field	Description
preapprovalKey	xs:string A preapproval key that identifies the preapproval requested. You can use this key in other Adaptive Payment requests to identify this preapproval.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the acknowledgement status.

ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.



FaultMessage Fields

Field	Description
error	common:ErrorData Detailed error information.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	<p>common:ErrorCategory</p> <p>The location where the error occurred.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • System – The system encountered errors; try again • Application – The application encountered errors; try again • Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	<p>common:ErrorParameter</p> <p>Represents contextual information about the error.</p>
severity	<p>common:ErrorSeverity</p> <p>The severity of the error encountered.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Error – Processing of the request was interrupted • Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	<p>common:AckCode</p> <p>Acknowledgement code. Possible values are:</p> <ul style="list-style-type: none"> • Success – Operation completed successfully • Failure – Operation failed • Warning – Warning • SuccessWithWarning – Operation completed successfully; however, there is a warning message • FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

Preapproval Examples Using NVP and CURL

These examples all use NVP for the data binding and CURL to deliver the HTTP request to the PayPal sandbox endpoint. Line breaks are provided for ease of reading; each CURL command is a single line and each request and response is a string without line breaks or extra whitespace.

Setting up a preapproval

In this example, the caller of the Preapproval API sets up a preapproval that is valid from July 17, 2009 through December 12, 2009. The maximum amount of the preapproval is \$1,500, of which a maximum of 30 payments are authorized with the largest payment not to exceed \$200. The sender need not include a PIN. The response contains the preapproval key that the caller can use in the Pay API operation.

NOTE: The sample code below uses the `insecure` setting to work around the certificate for testing in a sandbox environment. For actual implementations, you must specify the location of the certificate.

Request:

```
curl -s --insecure
-H "X-PAYPAL-SECURITY-USERID: api_username"
-H "X-PAYPAL-SECURITY-PASSWORD: api_password"
-H "X-PAYPAL-SECURITY-SIGNATURE: api_signature"
-H "X-PAYPAL-REQUEST-DATA-FORMAT: NV"
-H "X-PAYPAL-RESPONSE-DATA-FORMAT: NV"
-H "X-PAYPAL-APPLICATION-ID: app_id"
https://svcs.sandbox.paypal.com/AdaptivePayments/Preapproval -d
"cancelUrl=http://your_cancel_url
&currencyCode=USD
&endingDate=2009-12-13T08%3A00%3A00.000Z
&maxAmountPerPayment=200.00
&maxNumberOfPayments=30
&maxTotalAmountOfAllPayments=1500.00
&pinType=NOT_REQUIRED
&requestEnvelope.errorLanguage=en_US
&returnUrl=http://your_return_url
&startingDate=2009-07-13T07%3A00%3A00.000Z
&senderEmail=sender@domain
```

Response:

```
responseEnvelope.timestamp=2009-08-14T09%3A00%3A37.748-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=7967b2d03745a
&responseEnvelope.build=DEV
&preapprovalKey=PA-9JR04288NR0519129
```

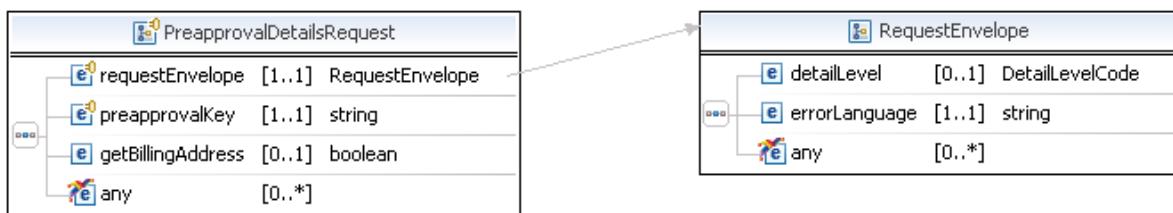
Preapproval Errors

Code	Message	Additional Information
500000	There is a system error	
520002	Internal error	
520003	User name/password is incorrect	
520006	This call is not defined in the database	
569016	Preapproval PIN functionality is not enabled	
569018	Preapproved payments have been disabled	
570026	The starting date is too far in the future	
579038	The date range between the start date and the end date is too wide	
579055	The limit for total amount of all payments is above PayPal's limit	
580001	Invalid request	
580022	Account already exists	Since 1.3.0
580024	The start date must be in the future	
580025	The start date must be earlier than the end date	
580027	The argument is unsupported	
580028	The URL <value> is malformed	
589023	If a fractional amount is rounded due to currency conversion, funds could be lost	
589039	The email address is invalid. It may not be registered in PayPal's system yet	

Use the PreapprovalDetails API operation to obtain information about an agreement between you and a sender for making payments on the sender's behalf.

PreapprovalDetailsRequest Message

The PreapprovalDetailsRequest message specifies the key of the preapproval agreement whose details you want to obtain.



PreapprovalDetailsRequest Fields

Field	Description
getBillingAddress	<p>xs:boolean</p> <p>(Optional) An option that lets you retrieve a list of billing addresses for the sender.</p> <ul style="list-style-type: none"> • true – Includes the billing address in the response • false – Omits the billing address from the response (default) <p>NOTE: This field is available only to API callers with advanced permission levels. For information, refer to the section Adaptive Payments Permission Levels.</p>
preapprovalKey	<p>xs:string</p> <p>(Required) A preapproval key that identifies the preapproval for which you want to retrieve details. The preapproval key is returned in the PreapprovalResponse message.</p>
requestEnvelope	<p>common:requestEnvelope</p> <p>(Required) Information common to each API operation, such as the language in which an error message is returned.</p>

PreapprovalDetails API Operation

PreapprovalDetailsResponse Message

RequestEnvelope Fields

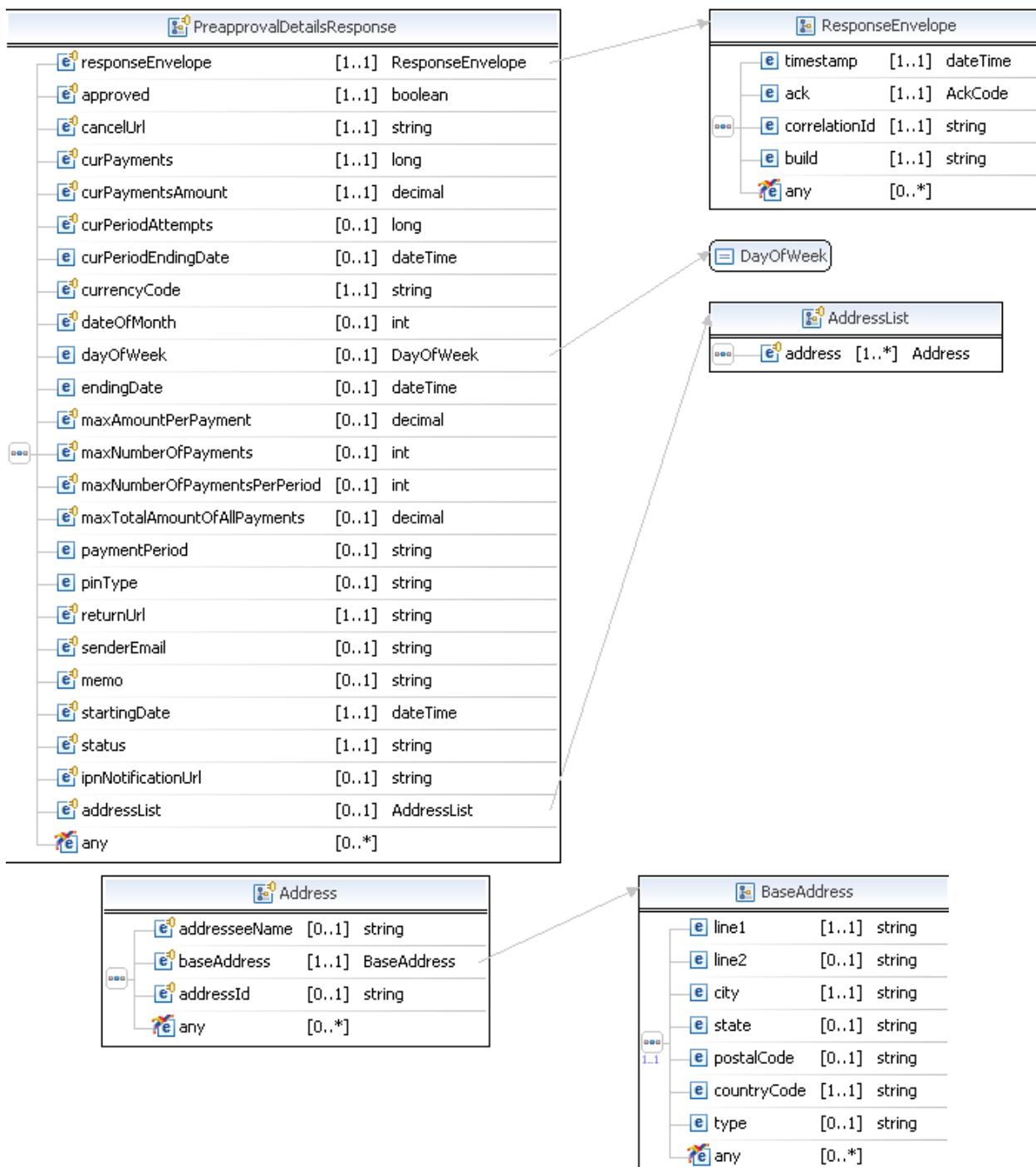
Field	Description
detailLevel	common:DetailLevelCode <i>(Optional)</i> The level of detail required by the client application for component such as Item, Transaction. Possible values are: <ul style="list-style-type: none">• ReturnAll – This value provides the maximum level of detail (default)
errorLanguage	xs:string <i>(Required)</i> The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported

PreapprovalDetailsResponse Message

The PreapprovalDetailsResponse message provides details about the requested preapproval.

PreapprovalDetails API Operation

PreapprovalDetailsResponse Message



PreapprovalDetails API Operation

PreapprovalDetailsResponse Message

PreapprovalDetailsResponse Fields

Field	Description
addresslist	ap:AddressList Returns a list of billing addresses.
approved	Whether the preapproval request was approved. Possible values are: <ul style="list-style-type: none">• true – The request was approved• false – The request was denied
cancelUrl	The URL to which the sender's browser is redirected if the sender decides to cancel the preapproval as requested. Use the preapproval key to identify the preapproval as follows: preapprovalKey=\${preapprovalKey}.
curPayments	The current number of payments by the sender for this preapproval.
curPaymentsAmount	The current total of payments by the sender for this preapproval.
curPeriodAttempts	The current number of payments by the sender this period for this preapproval.
curPeriodEndingDate	Ending date for the current period. Time is currently not supported.

Field	Description
currencyCode	<p>The currency code represented by the following values:</p> <ul style="list-style-type: none"> • Australian Dollar – AUD • Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none"> • Canadian Dollar – CAD • Czech Koruna – CZK • Danish Krone – DKK • Euro – EUR • Hong Kong Dollar – HKD • Hungarian Forint – HUF • Israeli New Sheqel – ILS • Japanese Yen – JPY • Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none"> • Mexican Peso – MXN • Norwegian Krone – NOK • New Zealand Dollar – NZD • Philippine Peso – PHP • Polish Zloty – PLN • Pound Sterling – GBP • Singapore Dollar – SGD • Swedish Krona – SEK • Swiss Franc – CHF • Taiwan New Dollar – TWD • Thai Baht – THB • U.S. Dollar – USD
dateOfMonth	The day of the month on which a monthly payment is to be made. A number between 1 and 31 indicates the day of the month. 0 indicates that the payment can be made on any day.
dayOfWeek	The day of the week that a weekly payment is to be made. Possible values are:
	<ul style="list-style-type: none"> • NO_DAY_SPECIFIED • SUNDAY • MONDAY • TUESDAY • WEDNESDAY • THURSDAY • FRIDAY • SATURDAY
endingDate	Last date for which the preapproval is valid. Time is currently not supported.
ipnNotificationUrl	The URL to which all IPN messages for this preapproval are sent.

PreapprovalDetails API Operation

PreapprovalDetailsResponse Message

Field	Description
maxAmountPerPayment	The preapproved maximum amount per payment.
maxNumberOfPayments	The preapproved maximum number of payments. This field is only returned if provided.
maxNumberOfPaymentsPerPeriod	The preapproved maximum number of payments per period.
maxTotalAmountOfAllPayments	The preapproved maximum total amount of all payments.
memo	A note about the preapproval.
paymentPeriod	The payment period. Possible values are: <ul style="list-style-type: none">● NO_PERIOD_SPECIFIED● DAILY● WEEKLY● BIWEEKLY● SEMIMONTHLY● MONTHLY● ANNUALLY
pinType	Whether a personal identification number (PIN) is required. It is one of the following values: <ul style="list-style-type: none">● NOT_REQUIRED – A PIN is not required● REQUIRED – A PIN is required
responseEnvelope	Common response information, including a timestamp and the acknowledgement status.
returnUrl	The URL to which the sender's browser is redirected after the sender approves the preapproval on paypal.com. Use the preapproval key to identify the preapproval as follows: preapprovalKey=\${preapprovalKey}.
senderEmail	Sender's email address. If not specified, the email address of the sender who logs in to approve the request becomes the email address associated with the preapproval key.
startDate	First date for which the preapproval is valid.
status	Whether this preapproval is active, represented by the following values: <ul style="list-style-type: none">● ACTIVE – The preapproval is active● CANCELED – The preapproval was explicitly canceled by the sender or by PayPal● DEACTIVED – The preapproval is not active; you can be reactivate it by resetting the personal identification number (PIN) or by contacting PayPal

AddressList Fields

Field	Description
AddressList	ap:Address Returns a billing address.

Address Fields

Field	Description
addresseeName	xs:string The name associated with the address.
baseAddress	common:BaseAddress Street address.
addressId	xs:string The ID associated with the address.

BaseAddress Fields

Field	Description
city	xs:string The city of the address.
countryCode	xs:string The country code of the address.
line1	xs:string The first line of the address.
line2	xs:string An second line of the street address.
postalCode	xs:string The postal code of the address.
state	xs:string The state for the address
type	xs:string. The type of address.

PreapprovalDetails API Operation

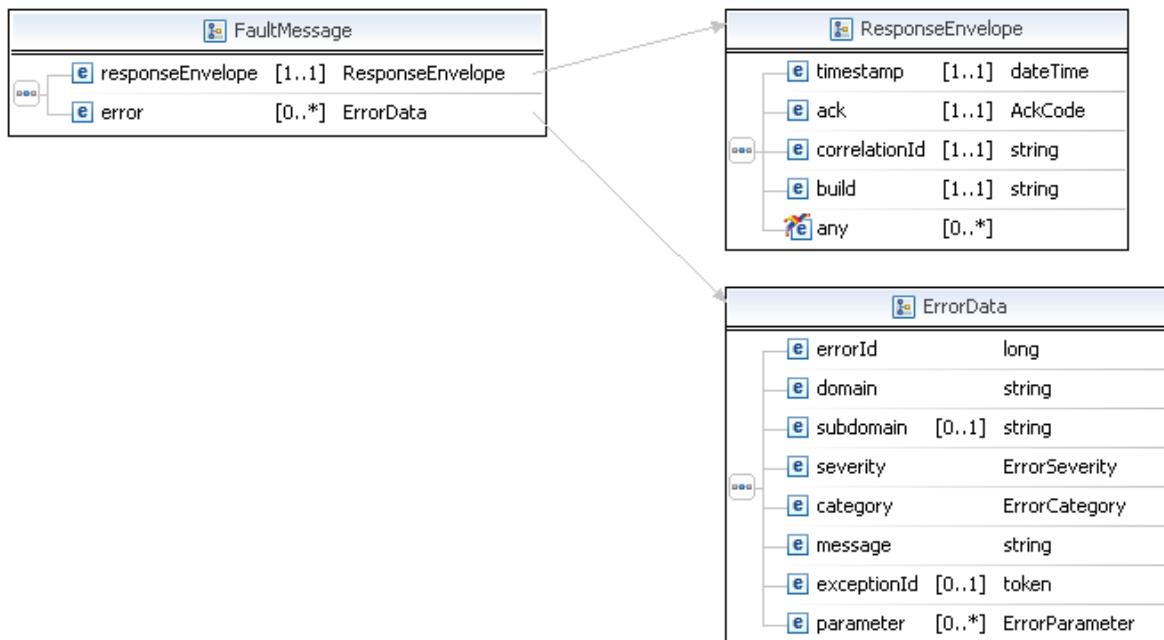
PPFault Message

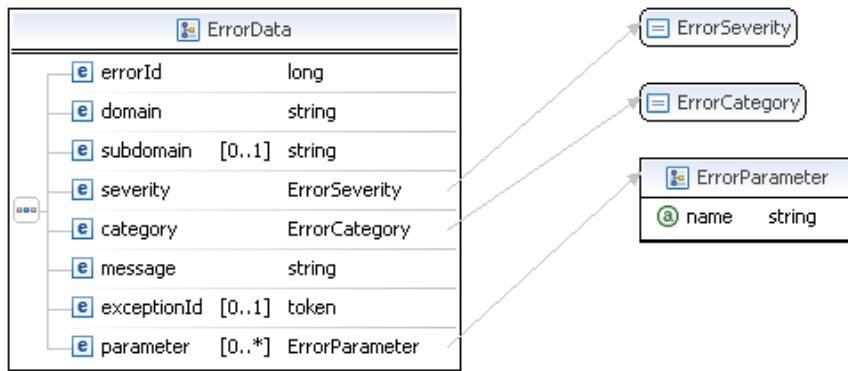
ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.





FaultMessage Fields

Field	Description
error	common:ErrorData Detailed error information.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	common:ErrorCategory The location where the error occurred. Possible values are: <ul style="list-style-type: none">• System – The system encountered errors; try again• Application – The application encountered errors; try again• Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	common:ErrorParameter Represents contextual information about the error.

PreapprovalDetails API Operation

PreapprovalDetails Examples Using NVP and CURL

Field	Description
severity	common:ErrorSeverity The severity of the error encountered. Possible values are: <ul style="list-style-type: none">• Error – Processing of the request was interrupted• Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

PreapprovalDetails Examples Using NVP and CURL

These examples all use NVP for the data binding and CURL to deliver the HTTP request to the PayPal sandbox endpoint. Line breaks are provided for ease of reading; each CURL command is a single line and each request and response is a string without line breaks or extra whitespace.

Obtaining information about a preapproval

In this example, the caller of the PreapprovalDetails API operation specifies a preapproval key. The result shows whether or not the status has been approved, which in this case is `false` because the person making the approval has not completed the process on PayPal.

NOTE: The sample code below uses the `insecure` setting to work around the certificate for testing in a sandbox environment. For actual implementations, you must specify the location of the certificate.

Request:

```
curl -s --insecure
-H "X-PAYPAL-SECURITY-USERID: api_username"
-H "X-PAYPAL-SECURITY-PASSWORD: api_password"
-H "X-PAYPAL-SECURITY-SIGNATURE: api_signature"
-H "X-PAYPAL-REQUEST-DATA-FORMAT: NV"
-H "X-PAYPAL-RESPONSE-DATA-FORMAT: NV"
-H "X-PAYPAL-APPLICATION-ID: app_id"
https://svcs.sandbox.paypal.com/AdaptivePayments/PreapprovalDetails -d
"preapprovalKey=PA-9JR04288NR0519129
&requestEnvelope.errorLanguage=en_US "
```

Response:

```
responseEnvelope.timestamp=2009-07-13T13%3A50%3A40.496-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=16dbc35f1aea7
&responseEnvelope.build=DEV
&approved=false
&cancelUrl=your_cancel_url
&curPayments=0
&curPaymentsAmount=0.00
&curPeriodAttempts=0
&currencyCode=USD
&dateOfMonth=0
&dayOfWeek=NO_DAY_SPECIFIED
&endingDate=2009-12-13T13%3A40%3A37.0-08%3A00&maxAmountPerPayment=200.00
&maxNumberOfPayments=30
&maxTotalAmountOfAllPayments=1500.00
&paymentPeriod=NO_PERIOD_SPECIFIED
&pinType=NOT_REQUIRED
&returnUrl=your_cancel_url
&startingDate=2009-07-13T13%3A40%3A37.0-07%3A00
&status=ACTIVE
```

Preapproval Details Errors

Code	Message	Additional Information
500000	There is a system error	
520002	Internal error	
520003	User name/password is incorrect	
520003	Authentication failed. API credentials are incorrect.	Since 1.6.0
520006	This call is not defined in the database	

PreapprovalDetails API Operation

Preapproval Details Errors

Code	Message	Additional Information
540031	You do not have permission to get these preapproval details	Since 1.6.0
550001	User is not allowed to perform this action	
580001	Invalid request	
580001	Invalid request: <value>	Since 1.6.0
589019	The preapproval key is invalid	
589019	The preapproval key <value> is invalid	Since 1.6.0

Use the CancelPreapproval API operation to handle the canceling of preapprovals. Preapprovals can be canceled regardless of the state they are in, such as active, expired, deactivated, and previously canceled.

CancelPreapprovalRequest Message

The CancelPreapprovalRequest message specifies which preapproval is to be canceled by identifying the preapproval with its the preapproval key.



CancelPreapprovalRequest Fields

The table below lists the fields for the CancelPreapprovalRequest message.

Field	Description
preapprovalKey	xs:string <i>(Required)</i> The preapproval key that identifies the preapproval to be canceled.
requestEnvelope	common:requestEnvelope <i>(Required)</i> Information common to each API operation, such as the language in which an error message is returned.

RequestEnvelope Fields

Field	Description
detailLevel	common:DetailLevelCode <i>(Optional)</i> The level of detail required by the client application for component such as Item, Transaction. Possible values are: <ul style="list-style-type: none">● ReturnAll – This value provides the maximum level of detail (default)

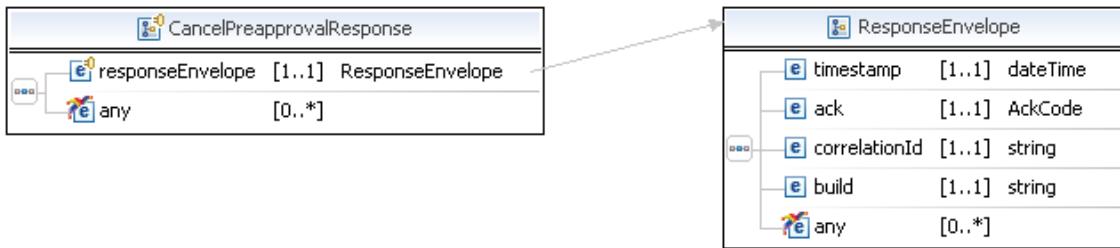
CancelPreapproval API Operation

CancelPreapprovalResponse Message

Field	Description
errorLanguage	xs:string <i>(Required)</i> The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported

CancelPreapprovalResponse Message

The CancelPreapprovalResponse message contains information returned in response to a CancelPreapprovalRequest message.



CancelPreapprovalResponse Fields

Field	Description
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

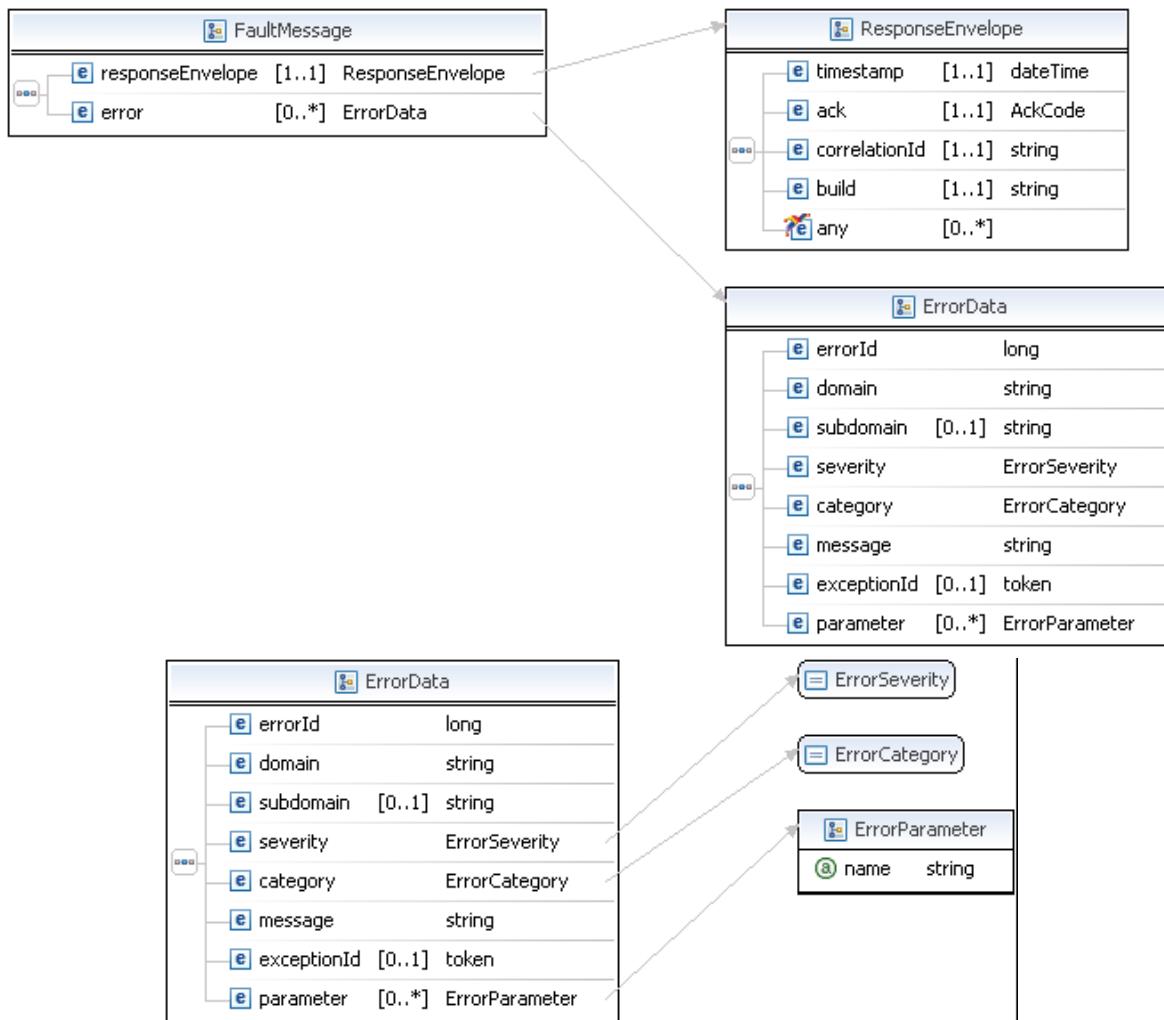
ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.

Field	Description
timestamp	The date on which the response was sent. The time is currently not supported.

PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.



CancelPreapproval API Operation

PPFault Message

FaultMessage Fields

Field	Description
error	common:ErrorData Detailed error information.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	common:ErrorCategory The location where the error occurred. Possible values are: <ul style="list-style-type: none">• System – The system encountered errors; try again• Application – The application encountered errors; try again• Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	common:ErrorParameter Represents contextual information about the error.
severity	common:ErrorSeverity The severity of the error encountered. Possible values are: <ul style="list-style-type: none">• Error – Processing of the request was interrupted• Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationID	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

CancelPreapproval Errors

Code	Message	Additional Information
520002	Internal error	
540031	You don't have permission to cancel this preapproval	
589019	The preapproval key is invalid	

Use the Refund API operation to refund all or part of a payment. You can specify the amount of the refund and identify the accounts to receive the refund by the payment key or tracking ID, and optionally, by transaction ID or the receivers of the original payment.

Refund API Overview

In a refund, the terms *sender* and *receiver* refer to sender and receivers of the original payment using the Pay API operation. When making a refund, the sender's account receives the refund and the receivers' accounts are the source of the refund. Refunds are made from one or more receivers to a sender.

To make a refund using the Refund API operation, you must:

- have made a Pay request for which payment you want to refund the payment, or you must have received the part of the payment you want to refund, and
- have permission to make a refund on behalf of the receiver, unless you are also the receiver; in other words, each receiver from whose account is the source of the refund must have granted you third-party access permission to call the Refund API operation.

NOTE: A receiver can grant you third-party access to make a refund by logging in to PayPal, choosing **API Access** on the Profile page, then clicking the link to [Grant API permission](#) and selecting **Refund** after clicking **Configure a custom API authorization**.

For each kind of refund, you must specify values for the following fields:

Common fields for all refunds

Field	Description
currencyCode	The currency code. You must specify the currency code that matches the currency code of the original payment unless you also specify the payment key.
requestEnvelope.errorLanguage	The code for the language in which errors are returned, which must be en_US.

You can refund a payment from all receivers associated with a payment by specifying the payment's payment key or tracking ID:

Refund API Operation

Refund API Overview

Refunds that apply to all receivers

Field	Description
payKey	The key used to create the payment that you want to refund
trackingID	The tracking ID associated with the payment that you want to refund

You can make a refund to the sender of a payment from specific receivers associated with a payment key or tracking ID. In this case, you must specify:

- either the payment key or tracking ID that identifies the original payment
- for each receiver making a refund, the receiver's email address
- for each receiver, the amount to refund to the sender from the receiver's account:

Additional fields for refunds of specific amounts to specific receivers

Field	Description
receiverList.receiver(<i>n</i>).email	The receiver's email address, where <i>n</i> is between 0 and 5 for a maximum of 6 receivers
receiverList.receiver(<i>n</i>).amount	Amount to refund from the receiver identified by email address

A payment can also be identified as a set of PayPal transactions between the sender and each specific receiver. You can refund the amount of the transaction by specifying the transaction ID:

Refunds that apply to a PayPal transaction associated with the payment

Field	Description
transactionID	A PayPal transaction ID associated with the receiver whose payment you want to refund to the sender

Refund Notifications

Notifications are sent after the refund is complete:

- PayPal sends an email to all PayPal account holders affected by the refund.
- PayPal sends an IPN message to the URL specified in the ipnNotificationUrl field of the Refund request.

Additional Notes About the Refund API Operation

1. Payments made using the Pay API operation cannot be refunded using the RefundTransaction API operation. You must use the Refund API operation or log in to PayPal to issue the refund.

2. A personal account holder cannot grant API access to a third party. A personal account holder must issue the refund by specifying the transaction ID, or by logging in to paypal.com to make the refund.
3. A payment made using the Pay API operation cannot be refunded by logging in to paypal.com if the fees cannot be refunded. In these cases, you must use the Refund API operation to issue the refund.
4. In the refund request, specify the payment using either the pay key, transaction ID, or tracking ID; it is not recommended to specify more than one of these arguments, unless they identify the same payment. If the arguments do not belong to the same payment, the transaction ID is used; next, it uses the payment key. If more than one way is specified, the tracking ID is never used.
5. With the Adaptive Payments Pay API operation, you can make payments to a person who does not have a PayPal account. In essence, you send a payment request that includes an email address for a receiver, and this email address is not linked to a registered PayPal account. The receiver receives an email notifying the receiver to create an account and claim the payment. Supporting features include email notifications and IPNs to the sender, receiver, and API caller, plus the ability to cancel unclaimed payments.

You can make such a payment in the same way that you make any other kind of payment; however, PayPal holds a payment to a receiver whose email address is not yet registered or confirmed until the receiver creates a PayPal account and confirms the email address. If a refund specifies a receiver whose email address is not yet registered or confirmed, the payment to the receiver is canceled.

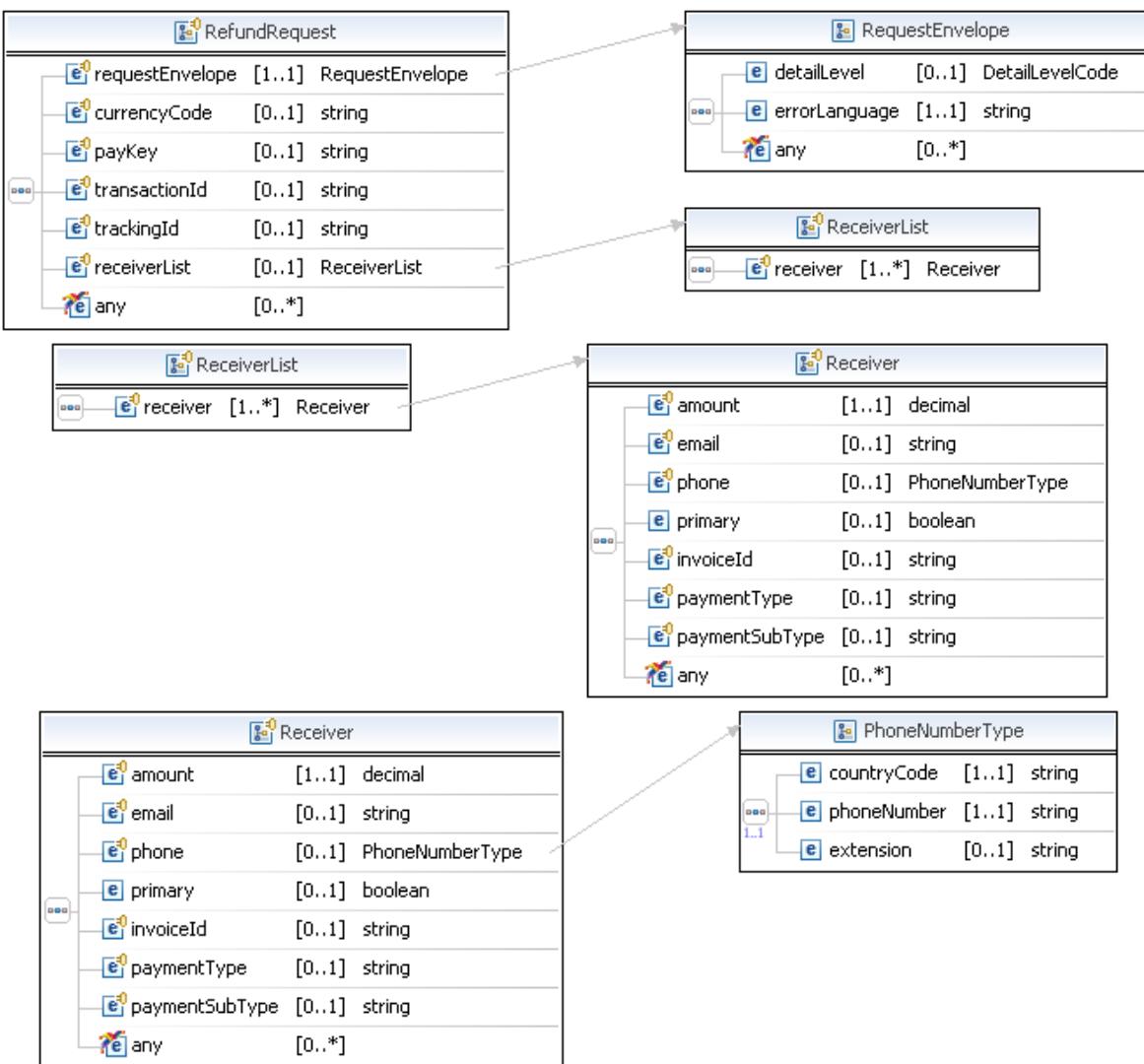
6. For chained payments, if the primary receiver's account has insufficient funds to cover the refund, an eCheck is issued, and the refund is put on hold until the check clears. No money will be moved from the secondary receivers' accounts until the primary receiver's fund are available using eCheck. Note that the Refund API operation does not support iACH.

RefundRequest Message

The `refundRequest` message specifies information about the refund and how it is to be allocated to PayPal account holders.

Refund API Operation

RefundRequest Message



RefundRequest Fields

You must specify a value for only one of the following fields to identify the part of the payment to refund: payKey, transactionId, or trackingId.

Field	Description
currencyCode	<p>(Required) The currency code. You must specify the currency code used for the original payment. You do not need to specify this field if you use a payKey to refund a complete transaction.</p> <p>Allowable values are:</p> <ul style="list-style-type: none"> • Australian Dollar – AUD • Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none"> • Canadian Dollar – CAD • Czech Koruna – CZK • Danish Krone – DKK • Euro – EUR • Hong Kong Dollar – HKD • Hungarian Forint – HUF • Israeli New Sheqel – ILS • Japanese Yen – JPY • Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none"> • Mexican Peso – MXN • Norwegian Krone – NOK • New Zealand Dollar – NZD • Philippine Peso – PHP • Polish Zloty – PLN • Pound Sterling – GBP • Singapore Dollar – SGD • Swedish Krona – SEK • Swiss Franc – CHF • Taiwan New Dollar – TWD • Thai Baht – THB • U.S. Dollar – USD
payKey	<p>xs:string</p> <p>(Optional) The key used to create the payment that you want to refund.</p>
receiverList	<p>ap:ReceiverList</p> <p>(Optional) Information about the receivers of the payment that you want to refund; the payment is identified by payment key, tracking ID, or transaction ID. Specify each receiver to whom you want to issue a refund. You can specify the refund amount for specific receivers. If you do not specify any receivers, all receivers will be fully refunded if you specify the payment key or tracking ID; otherwise, only the affected receiver will be fully refunded if you specify the transaction ID.</p>

Refund API Operation

RefundRequest Message

Field	Description
requestEnvelope	common:requestEnvelope <i>(Required)</i> Information common to each API operation, such as the language in which an error message is returned.
transactionId	xs:string <i>(Optional)</i> The PayPal transaction ID associated with the payment that you want to refund.
trackingId	xs:string <i>(Optional)</i> The tracking ID associated with the payment that you want to refund.

ReceiverList Fields

Field	Description
receiver	ap:Receiver <i>(Required)</i> Receiver is the party whose account is credited.

Receiver Fields

Field	Description
amount	xs:decimal <i>(Required)</i> Amount to be credited to the receiver's account.
email	xs:string <i>(Required)</i> Receiver's email address. Maximum length: 127 characters
invoiceId	xs:string <i>(Optional)</i> This field is not used. Maximum length: 127 characters
paymentType	xs:string <i>(Optional)</i> This field is not used.
paymentSubType	xs:string <i>(Optional)</i> This field is not used.
phone	common:PhoneNumberType <i>(Optional)</i> A type to specify the receiver's phone number. The PayRequest must pass either an email address or a phone number as the payment receiver.

Field	Description
primary	<p>xs:boolean</p> <p>(Optional) Whether this receiver is the primary receiver, which makes this a refund for a chained payment. You can specify at most one primary receiver. Omit this field for refunds for simple and parallel payments.</p> <p>Allowable values are:</p> <ul style="list-style-type: none"> • true – Primary receiver • false – Secondary receiver (default)

PhoneNumberType Fields

Field	Description
countryCode	<p>xs:string</p> <p>(Required) Telephone country code.</p>
phoneNumber	<p>xs:string</p> <p>(Required) Telephone number.</p>
extension	<p>xs:string</p> <p>(Optional) Telephone extension.</p>

RequestEnvelope Fields

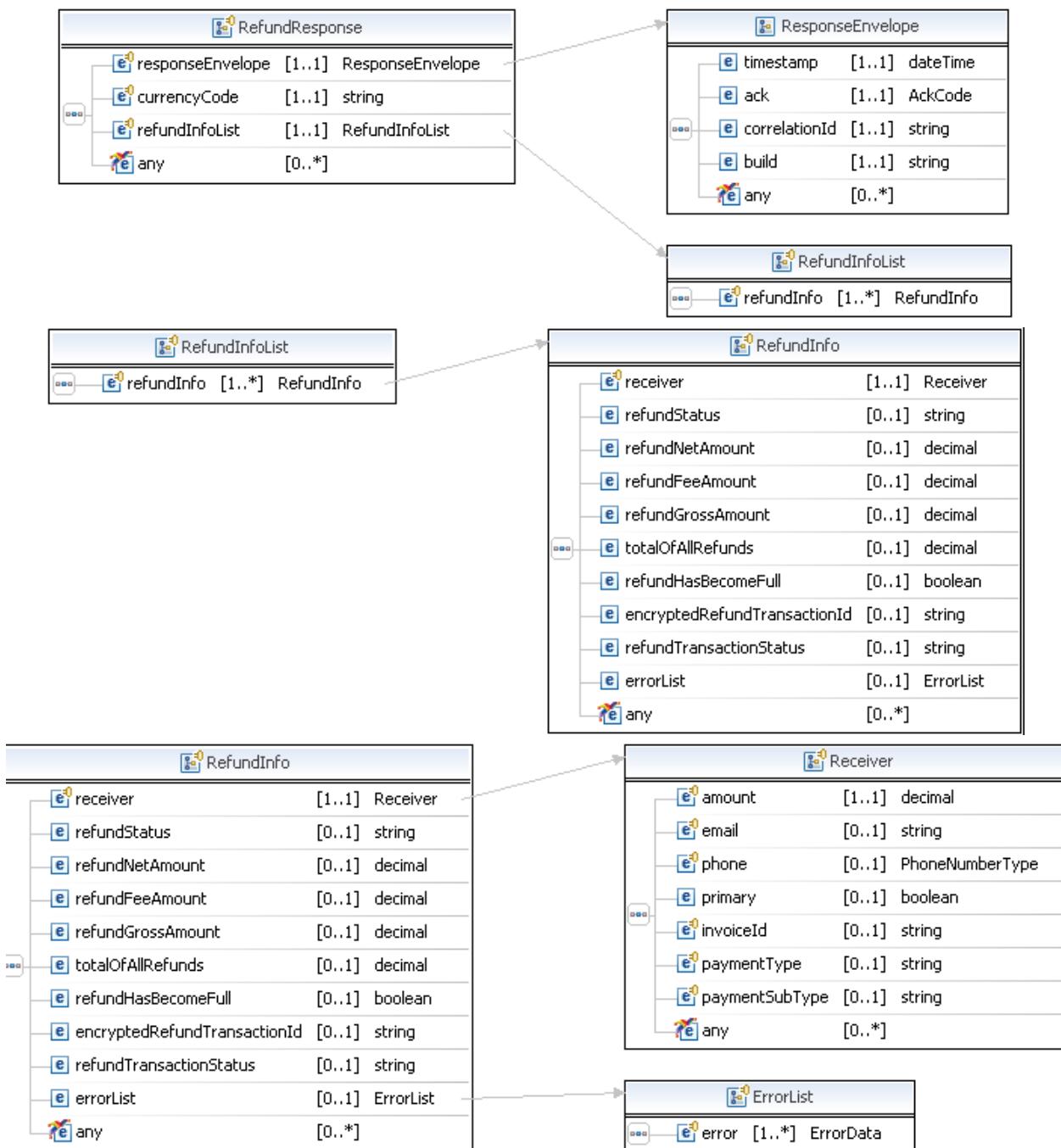
Field	Description
detailLevel	<p>common:DetailLevelCode</p> <p>(Optional) The level of detail required by the client application for component such as Item, Transaction. Possible values are:</p> <ul style="list-style-type: none"> • ReturnAll – This value provides the maximum level of detail (default)
errorLanguage	<p>xs:string</p> <p>(Required) The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported</p>

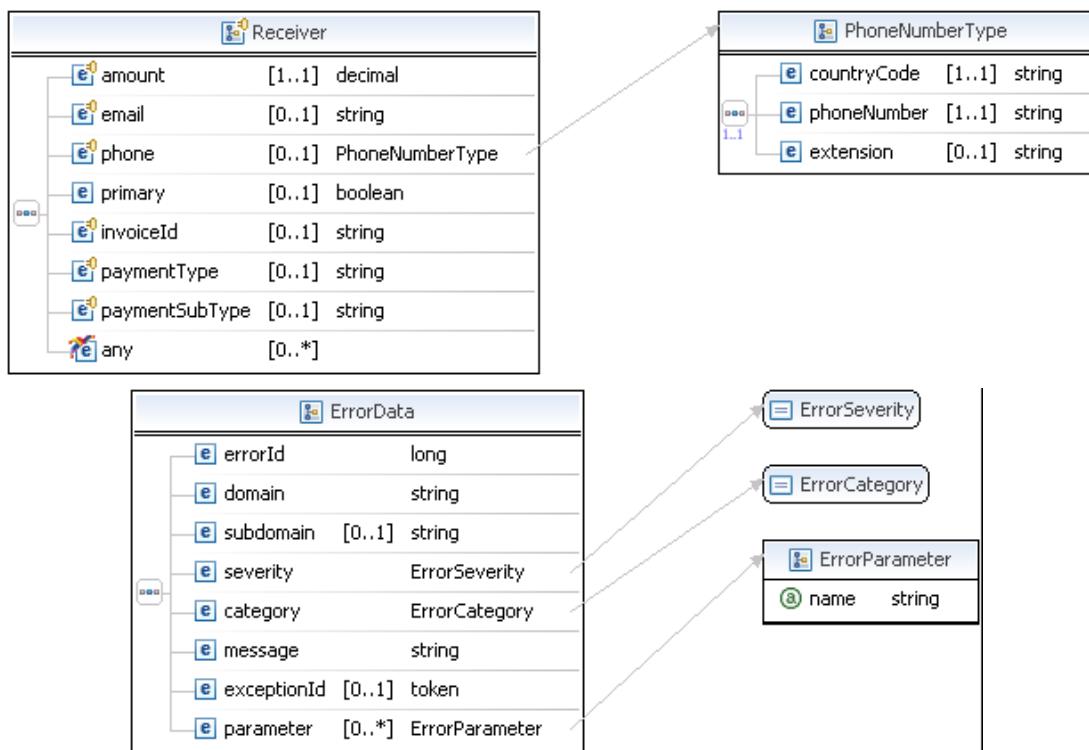
Refund Response Message

The refundResponse message contains status information about a refund request.

Refund API Operation

Refund Response Message





RefundResponse Fields

Field	Description
currencyCode	<p>The currency code represented by one of the following values:</p> <ul style="list-style-type: none"> • Australian Dollar – AUD • Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none"> • Canadian Dollar – CAD • Czech Koruna – CZK • Danish Krone – DKK • Euro – EUR • Hong Kong Dollar – HKD • Hungarian Forint – HUF • Israeli New Sheqel – ILS • Japanese Yen – JPY • Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none"> • Mexican Peso – MXN • Norwegian Krone – NOK • New Zealand Dollar – NZD • Philippine Peso – PHP • Polish Zloty – PLN • Pound Sterling – GBP • Singapore Dollar – SGD • Swedish Krona – SEK • Swiss Franc – CHF • Taiwan New Dollar – TWD • Thai Baht – THB • U.S. Dollar – USD
refundInfoList	<p>ap:RefundInfoList</p> <p>List of refunds associated with the payment.</p>
responseEnvelope	<p>common:ResponseEnvelope</p> <p>Common response information, including a timestamp and the response acknowledgement status.</p>

RefundInfoList Fields

Field	Description
refundInfo	<p>ap:RefundInfo</p> <p>Represents the refund attempt made to a receiver of a PayRequest.</p>

RefundInfo Fields

Field	Description
encryptedRefundTransacti onId	<p>xs:string</p> <p>The PayPal transaction ID for this refund.</p> <p>NOTE: This ID is different than the transaction ID for the original payment.</p>
errorList	<p>ap:ErrorList</p> <p>Information about why a refund failed.</p>
receiver	<p>ap:Receiver</p> <p>Information about the receiver of the refund.</p>
refundFeeAmount	<p>xs:decimal</p> <p>Amount of fees that have been refunded.</p>
refundGrossAmount	<p>xs:decimal</p> <p>Gross amount of the refund, including the total amount from this receiver and fees.</p>
refundHasBecomeFull	<p>xs:boolean</p> <p>Whether you have refunded the total amount of the payment.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • true – Entire amount of the payment has been refunded • false – Entire amount of the payment has not been refunded <p>NOTE: This field is only available when a partial refund has been requested.</p>
refundNetAmount	<p>xs:decimal</p> <p>The total amount of the refund from this receiver.</p>

Refund API Operation

Refund Response Message

Field	Description
refundStatus	<p>xs:string</p> <p>Status of the refund. It is one of the following values:</p> <ul style="list-style-type: none">• REFUNDED – Refund successfully completed• REFUNDED_PENDING – Refund awaiting transfer of funds; for example, a refund paid by eCheck.• NOT_PAID – Payment was never made; therefore, it cannot be refunded.• ALREADY_REVERSED_OR_REFUNDED – Request rejected because the refund was already made, or the payment was reversed prior to this request.• NO_API_ACCESS_TO_RECEIVER – Request cannot be completed because you do not have third-party access from the receiver to make the refund.• REFUND_NOT_ALLOWED – Refund is not allowed.• INSUFFICIENT_BALANCE – Request rejected because the receiver from which the refund is to be paid does not have sufficient funds or the funding source cannot be used to make a refund.• AMOUNT_EXCEEDS_REFUNDABLE – Request rejected because you attempted to refund more than the remaining amount of the payment; call the PaymentDetails API operation to determine the amount already refunded.• PREVIOUS_REFUND_PENDING – Request rejected because a refund is currently pending for this part of the payment• NOT_PROCESSED – Request rejected because it cannot be processed at this time• REFUND_ERROR – Request rejected because of an internal error• PREVIOUS_REFUND_ERROR – Request rejected because another part of this refund caused an internal error.
refundTransactionStatus	<p>xs:string</p> <p>The status of the refund transaction.</p> <p>Possible values are:</p> <ul style="list-style-type: none">• UNKNOWN – The status of the refund process is unknown.• COMPLETED – The refund process has been completed.
totalOfAllRefunds	<p>xs:decimal</p> <p>The total of all refunds actually paid; does not include pending amounts</p> <p>NOTE: This field is only available when a partial refund has been requested.</p>

ErrorData Fields

Field	Description
category	<p>common:ErrorCategory</p> <p>The location where the error occurred.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • System – The system encountered errors; try again • Application – The application encountered errors; try again • Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	<p>common:ErrorParameter</p> <p>Represents contextual information about the error.</p>
severity	<p>common:ErrorSeverity</p> <p>The severity of the error encountered.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Error – Processing of the request was interrupted • Warning – Processing of the request was completed
subdomain	This field is not used.

Receiver Fields

Field	Description
amount	<p>xs:decimal</p> <p>Amount to be refunded to the receiver.</p>
email	<p>xs:string</p> <p>Receiver's email address.</p> <p>Maximum length: 127 characters</p>
invoiceId	<p>xs:string</p> <p>This field is not used.</p> <p>Maximum length: 127 characters</p>
paymentType	<p>xs:string</p> <p>This field is not used.</p>
paymentSubType	<p>xs:string</p> <p>This field is not used.</p>

Refund API Operation

Refund Response Message

Field	Description
phone	common:PhoneNumberType The receiver's phone number.
primary	xs:boolean Whether this receiver is the primary receiver, which makes this a refund for a chained payment. If this field is set to false, this is a refund for a simple or parallel payment. Possible values are: <ul style="list-style-type: none">• true – Primary receiver (chained payment)• false – Secondary receiver (simple/parallel payment)

PhoneNumberType Response Fields

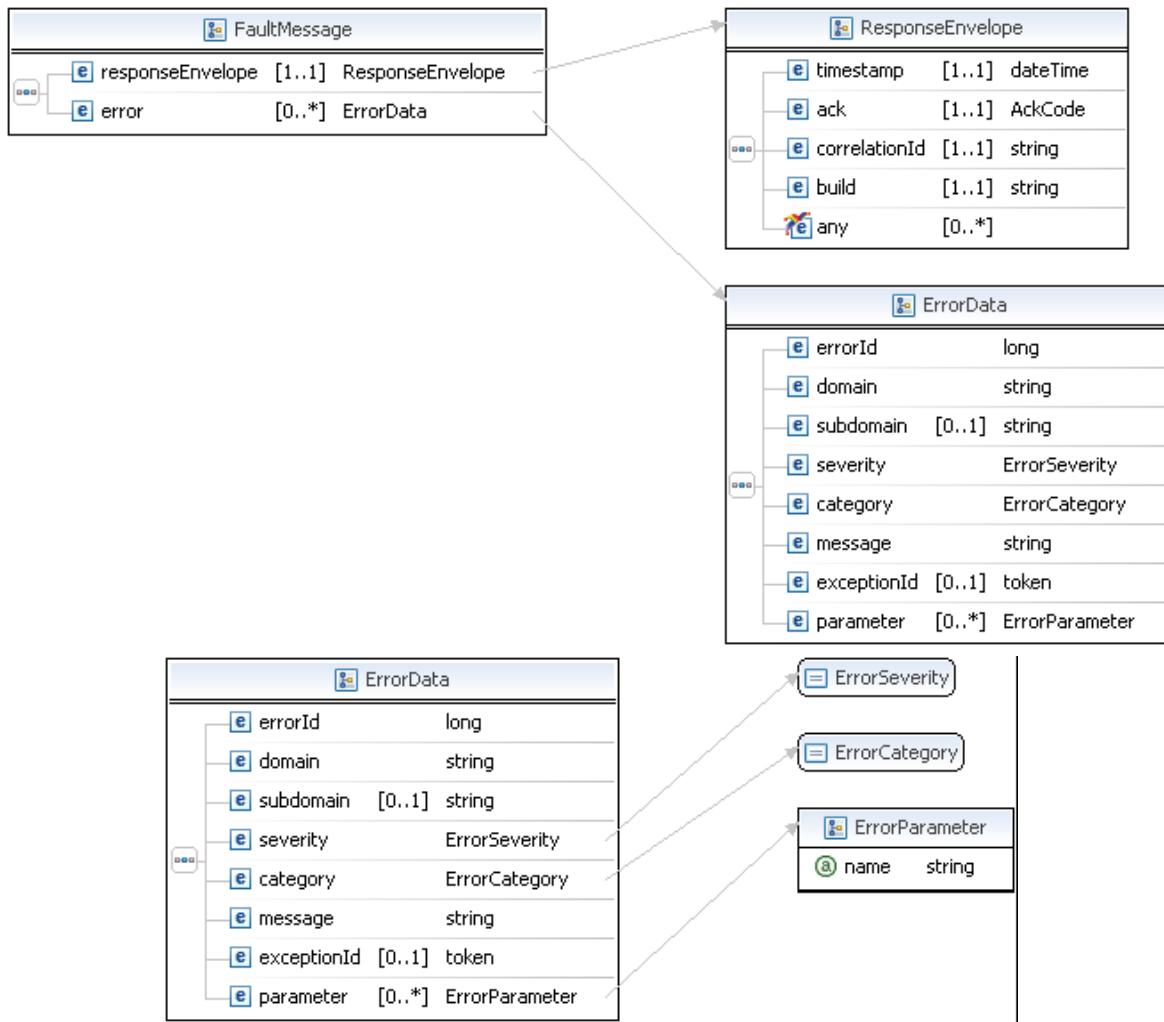
Field	Description
countryCode	xs:string Telephone country code.
phoneNumber	xs:string Telephone number.
extension	xs:string Telephone extension.

ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.



FaultMessage Fields

Field	Description
<code>error</code>	<code>common:ErrorData</code> Detailed error information.
<code>responseEnvelope</code>	<code>common:ResponseEnvelope</code> Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	<p>common:ErrorCategory</p> <p>The location where the error occurred.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • System – The system encountered errors; try again • Application – The application encountered errors; try again • Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	<p>common:ErrorParameter</p> <p>Represents contextual information about the error.</p>
severity	<p>common:ErrorSeverity</p> <p>The severity of the error encountered.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Error – Processing of the request was interrupted • Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	<p>common:AckCode</p> <p>Acknowledgement code. Possible values are:</p> <ul style="list-style-type: none"> • Success – Operation completed successfully • Failure – Operation failed • Warning – Warning • SuccessWithWarning – Operation completed successfully; however, there is a warning message • FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

Refund Examples Using NVP and CURL

These examples all use NVP for the data binding and CURL to deliver the HTTP request to the PayPal sandbox endpoint. Line breaks are provided for ease of reading; each CURL command is a single line and each request and response is a string without line breaks or extra whitespace.

Using the payment key to refund an entire payment

In this example, the caller of the Refund API operation specifies a payment key. The total amount from each receiver is refunded to the sender.

NOTE: The sample code below uses the `insecure` setting to work around the certificate for testing in a sandbox environment. For actual implementations, you must specify the location of the certificate.

Request:

```
curl -s --insecure  
-H "X-PAYPAL-SECURITY-USERID: api_username"  
-H "X-PAYPAL-SECURITY-PASSWORD: api_password"  
-H "X-PAYPAL-SECURITY-SIGNATURE: api_signature"  
-H "X-PAYPAL-REQUEST-DATA-FORMAT: NV"  
-H "X-PAYPAL-RESPONSE-DATA-FORMAT: NV"  
-H "X-PAYPAL-APPLICATION-ID: your_app_id"  
https://svcs.sandbox.paypal.com/AdaptivePayments/Refund -d  
"requestEnvelope.errorLanguage=en_US  
&payKey=AP-95V43510SV018561T"
```

Response:

Refund API Operation

Refund Errors

```
responseEnvelope.timestamp=2009-08-14T09%3A00%3A37.748-07%3A00
&responseEnvelope.ack=Success
&responseEnvelope.correlationId=7967b2d03745a
&responseEnvelope.build=DEV
&currencyCode=USD
&refundInfoList.refundInfo(0).receiver.amount=3.15
&refundInfoList.refundInfo(0).receiver.email=receiver1@domain
&refundInfoList.refundInfo(0).refundStatus=REFUNDED
&refundInfoList.refundInfo(0).refundNetAmount=2.81
&refundInfoList.refundInfo(0).refundFeeAmount=0.34
&refundInfoList.refundInfo(0).refundGrossAmount=3.15
&refundInfoList.refundInfo(0).totalOfAllRefunds=3.15
&refundInfoList.refundInfo(0).refundHasBecomeFull=true
&refundInfoList.refundInfo(0).encryptedRefundTransactionId=48K11199GC160...
&refundInfoList.refundInfo(0).refundTransactionStatus=COMPLETED
&refundInfoList.refundInfo(1).receiver.amount=1.78
&refundInfoList.refundInfo(1).receiver.email=receiver2@domain
&refundInfoList.refundInfo(1).refundStatus=REFUNDED
&refundInfoList.refundInfo(1).refundNetAmount=1.43
&refundInfoList.refundInfo(1).refundFeeAmount=0.35
&refundInfoList.refundInfo(1).refundGrossAmount=1.78
&refundInfoList.refundInfo(1).totalOfAllRefunds=1.78
&refundInfoList.refundInfo(1).refundHasBecomeFull=true
&refundInfoList.refundInfo(1).encryptedRefundTransactionId=40Y76985BU687...
&refundInfoList.refundInfo(1).refundTransactionStatus=COMPLETED
```

Refund Errors

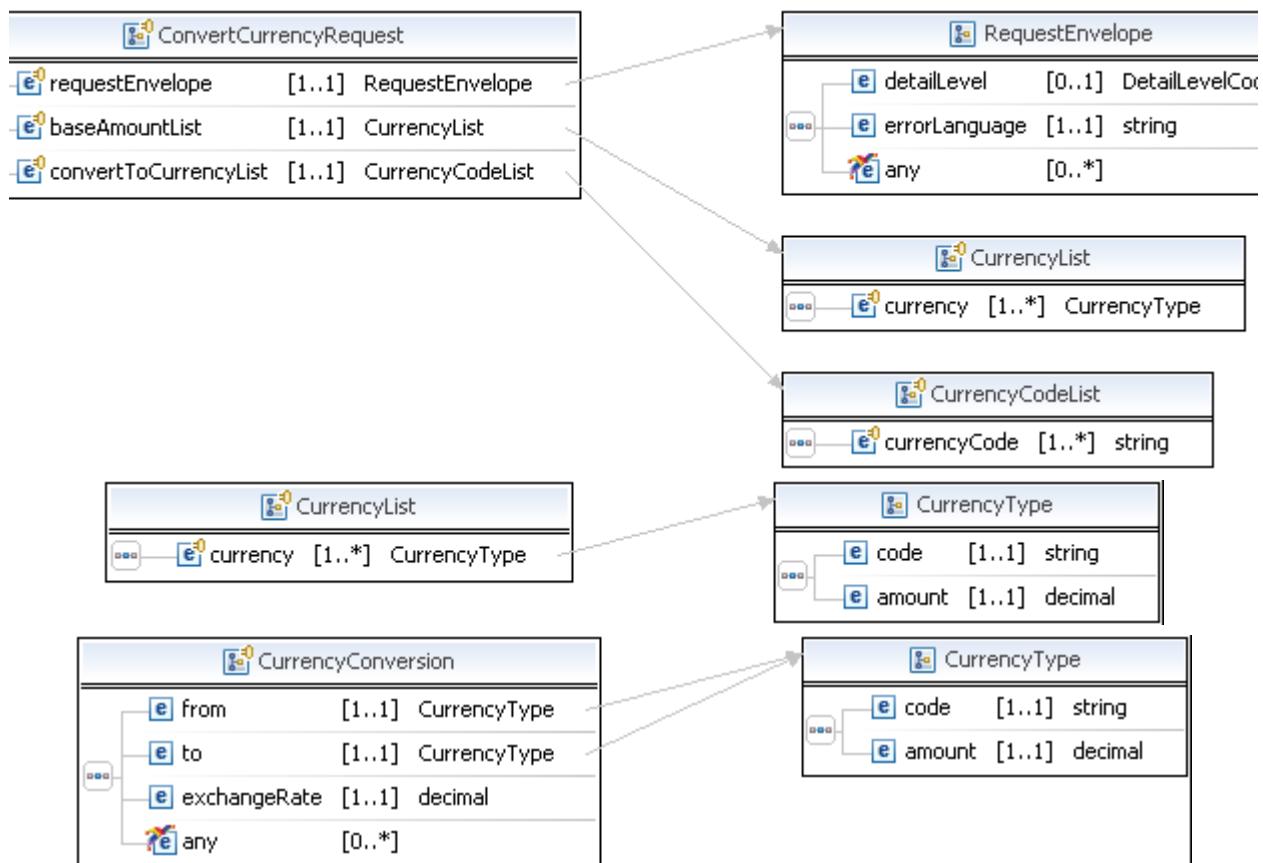
Code	Message	Additional Information
500000	There is a system error	
520002	Internal error	
520003	User name/password is incorrect	
520006	This call is not defined in the database	
520008	You do not have a verified ACH	
520009	Account is restricted	
520010	The account for the counter party is locked or inactive	
560011	Because a complaint case exists on this transaction, only a refund of the full or full remaining amount of the transaction can be issued	
560016	You cannot do a partial refund on this transaction	

Code	Message	Additional Information
560018	You cannot refund this type of transaction	
570012	Cannot do a full refund after a partial refund	
570013	The partial refund amount must be less than or equal to the original transaction amount	
570014	The partial refund amount must be less than or equal to the remaining amount	
570015	This transaction has already been fully refunded	
570017	You are over the time limit to perform a refund on this transaction	
579051	The tracking ID is invalid	
580001	Invalid request	
580022	Invalid request parameter: payKey with value <value>	
580027	The argument is unsupported	
589018	The pay key is valid	
589023	If a fractional amount is rounded due to currency conversion, funds could be lost	
580029	A pay key, transaction ID, or tracking ID is missing	Since 1.6.0
589036	The currency code is missing or does not match the code in the payment request	
589037	The refund's receiver was not part of the payment request	
589038	The refund has to be both less than or equal to the original payment request and greater than zero	
589041	All receivers must give full refunds for this type of chained payment	
589052	The transaction ID is invalid	
589061	The receiver <value> is invalid for this refund	Since 1.6.0

Use the ConvertCurrency API operation to obtain the current foreign exchange (FX) rate for a specific amount and currency.

ConvertCurrencyRequest Message

The ConvertCurrencyRequest message enables you to have your application get the exchange rate for a list of amounts. This API operation does not affect PayPal balances.



ConvertCurrencyRequest Fields

The table below describes the fields for the ConvertCurrencyRequest message.

ConvertCurrency API Operation

ConvertCurrencyRequest Message

Field	Description
baseAmountList	ap:CurrencyList <i>(Required)</i> A list of amounts with associated currencies to be converted.
convertToCurrencyList	ap:CurrencyCodeList <i>(Required)</i> A list of currencies to convert to.
requestEnvelope	common:RequestEnvelope <i>(Required)</i> Information common to each API operation, such as the language in which an error message is returned

CurrencyType Fields

Field	Description
amount	xs:decimal <i>(Required)</i> The amount to be converted.

Field	Description
code	<p>xs:string</p> <p>(Required) The currency code. Allowable values are:</p> <ul style="list-style-type: none"> • Australian Dollar – AUD • Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none"> • Canadian Dollar – CAD • Czech Koruna – CZK • Danish Krone – DKK • Euro – EUR • Hong Kong Dollar – HKD • Hungarian Forint – HUF • Israeli New Sheqel – ILS • Japanese Yen – JPY • Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none"> • Mexican Peso – MXN • Norwegian Krone – NOK • New Zealand Dollar – NZD • Philippine Peso – PHP • Polish Zloty – PLN • Pound Sterling – GBP • Singapore Dollar – SGD • Swedish Krona – SEK • Swiss Franc – CHF • Taiwan New Dollar – TWD • Thai Baht – THB • U.S. Dollar – USD

ConvertCurrency API Operation

ConvertCurrencyRequest Message

CurrencyList Fields

Field	Description
currency	<p>ap:CurrencyType</p> <p>(Required) The currency code. Allowable values are:</p> <ul style="list-style-type: none">• Australian Dollar – AUD• Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none">• Canadian Dollar – CAD• Czech Koruna – CZK• Danish Krone – DKK• Euro – EUR• Hong Kong Dollar – HKD• Hungarian Forint – HUF• Israeli New Sheqel – ILS• Japanese Yen – JPY• Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none">• Mexican Peso – MXN• Norwegian Krone – NOK• New Zealand Dollar – NZD• Philippine Peso – PHP• Polish Zloty – PLN• Pound Sterling – GBP• Singapore Dollar – SGD• Swedish Krona – SEK• Swiss Franc – CHF• Taiwan New Dollar – TWD• Thai Baht – THB• U.S. Dollar – USD

CurrencyCodeList Fields

Field	Description
currencyCode	<p>xs:string</p> <p>(Required) The currency code. Allowable values are:</p> <ul style="list-style-type: none"> • Australian Dollar – AUD • Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none"> • Canadian Dollar – CAD • Czech Koruna – CZK • Danish Krone – DKK • Euro – EUR • Hong Kong Dollar – HKD • Hungarian Forint – HUF • Israeli New Sheqel – ILS • Japanese Yen – JPY • Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none"> • Mexican Peso – MXN • Norwegian Krone – NOK • New Zealand Dollar – NZD • Philippine Peso – PHP • Polish Zloty – PLN • Pound Sterling – GBP • Singapore Dollar – SGD • Swedish Krona – SEK • Swiss Franc – CHF • Taiwan New Dollar – TWD • Thai Baht – THB • U.S. Dollar – USD

RequestEnvelope Fields

Field	Description
detailLevel	<p>common:DetailLevelCode</p> <p>(Optional) The level of detail required by the client application for component such as Item, Transaction. Possible values are:</p> <ul style="list-style-type: none"> • ReturnAll – This value provides the maximum level of detail (default)

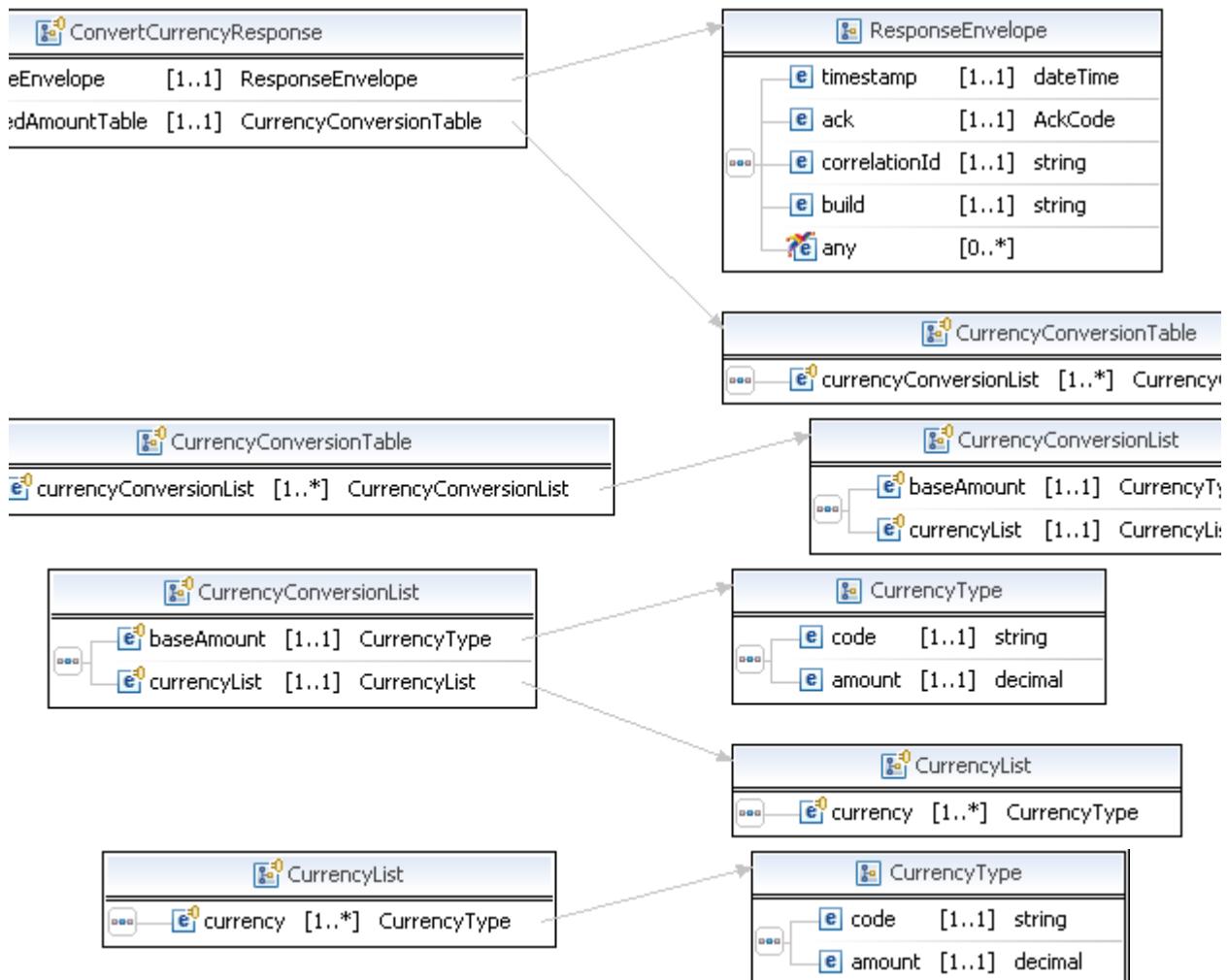
ConvertCurrency API Operation

ConvertCurrencyResponse Message

Field	Description
errorLanguage	xs:string <i>(Required)</i> The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported

ConvertCurrencyResponse Message

The ConvertCurrencyResponse message returns the following information to the application in response to the ConvertCurrencyRequest message.



ConversionCurrencyResponse Fields

Field	Description
estimatedAmountTable	ap:CurrencyConversionTable A list of the requested amounts converted to the requested currencies.
responseEnvelope	common:responseEnvelope Information common to each API operation, such as the language in which an error message is returned.

CurrencyConversionTable Fields

Field	Description
currencyConversionList	ap:CurrencyConversionList The list of converted currencies.

CurrencyConversionList Fields

Field	Description
baseAmount	ap:CurrencyType The base amount and currency code.
currencyList	ap:CurrencyList The list of amounts converted to the requested currencies, and the currency codes.

CurrencyType Fields

Field	Description
amount	xs:decimal The converted amount.

ConvertCurrency API Operation

ConvertCurrencyResponse Message

Field	Description
code	<p>xs:string</p> <p>The currency code for the converted amount. Possible values are:</p> <ul style="list-style-type: none">• Australian Dollar – AUD• Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none">• Canadian Dollar – CAD• Czech Koruna – CZK• Danish Krone – DKK• Euro – EUR• Hong Kong Dollar – HKD• Hungarian Forint – HUF• Israeli New Sheqel – ILS• Japanese Yen – JPY• Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none">• Mexican Peso – MXN• Norwegian Krone – NOK• New Zealand Dollar – NZD• Philippine Peso – PHP• Polish Zloty – PLN• Pound Sterling – GBP• Singapore Dollar – SGD• Swedish Krona – SEK• Swiss Franc – CHF• Taiwan New Dollar – TWD• Thai Baht – THB• U.S. Dollar – USD

CurrencyList Fields

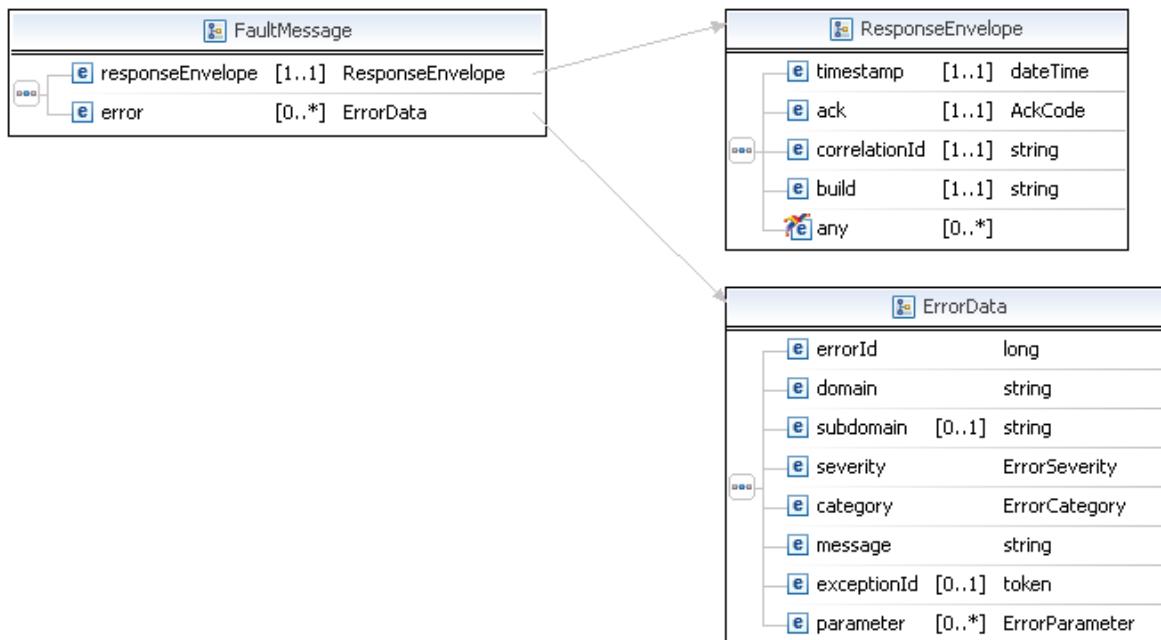
Field	Description
currency	<p>ap:CurrencyType</p> <p>The list of converted amounts with their target currency codes.</p>

ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

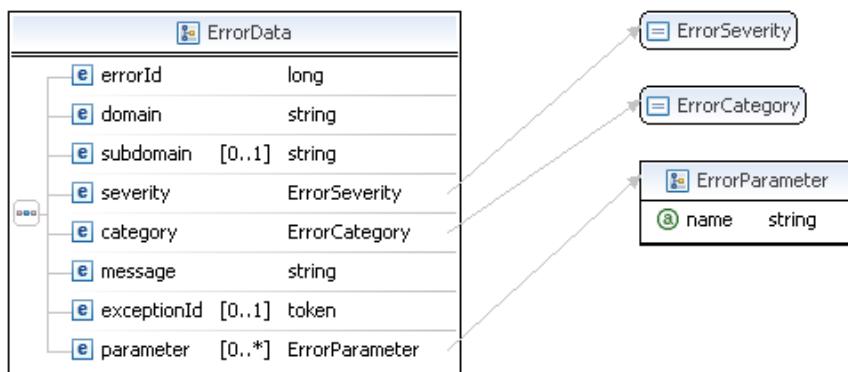
PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.



ConvertCurrency API Operation

PPFault Message



FaultMessage Fields

Field	Description
error	common:ErrorData Detailed error information.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	common:ErrorCategory The location where the error occurred. Possible values are: <ul style="list-style-type: none">• System – The system encountered errors; try again• Application – The application encountered errors; try again• Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	common:ErrorParameter Represents contextual information about the error.

Field	Description
severity	<p>common:ErrorSeverity</p> <p>The severity of the error encountered.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • Error – Processing of the request was interrupted • Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	<p>common:AckCode</p> <p>Acknowledgement code. Possible values are:</p> <ul style="list-style-type: none"> • Success – Operation completed successfully • Failure – Operation failed • Warning – Warning • SuccessWithWarning – Operation completed successfully; however, there is a warning message • FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

ConvertCurrency Errors

Code	Message	Additional Information
520002	Internal error	
580022	Your request was invalid. Check the parameter of the error to see which is invalid	
580027	The argument is unsupported	

Use the GetFundingPlans API operation to determine the funding sources that are available for a specified payment, identified by its key, which takes into account the preferences and country of the receiver as well as the payment amount. You must be both the sender of the payment and the caller of this API operation.

GetFundingPlans Overview

To obtain the funding plans for a payment, specify values for the following fields:

Required fields

Field	Description
payKey	The key used to create the payment whose funding sources you want to determine.
requestEnvelope.errorLanguage	The code for the language in which errors are returned, which must be en_US.

Additional Notes About the Pay API Operation

1. The GetFundingPlans API operation is only available for implicitly approved payments, meaning you must be both the sender and the caller of the Pay API operation identified by the pay key.

GetFundingPlansRequest Message



GetFundingPlans API Operation

GetFundingPlansResponse Message

GetFundingPlansRequest Fields

Field	Description
payKey	<code>xs:string</code> <i>(Required)</i> The key used to create the payment whose funding sources you want to determine.
requestEnvelope	<code>common:requestEnvelope</code> <i>(Required)</i> Information common to each API operation, such as the language in which an error message is returned.

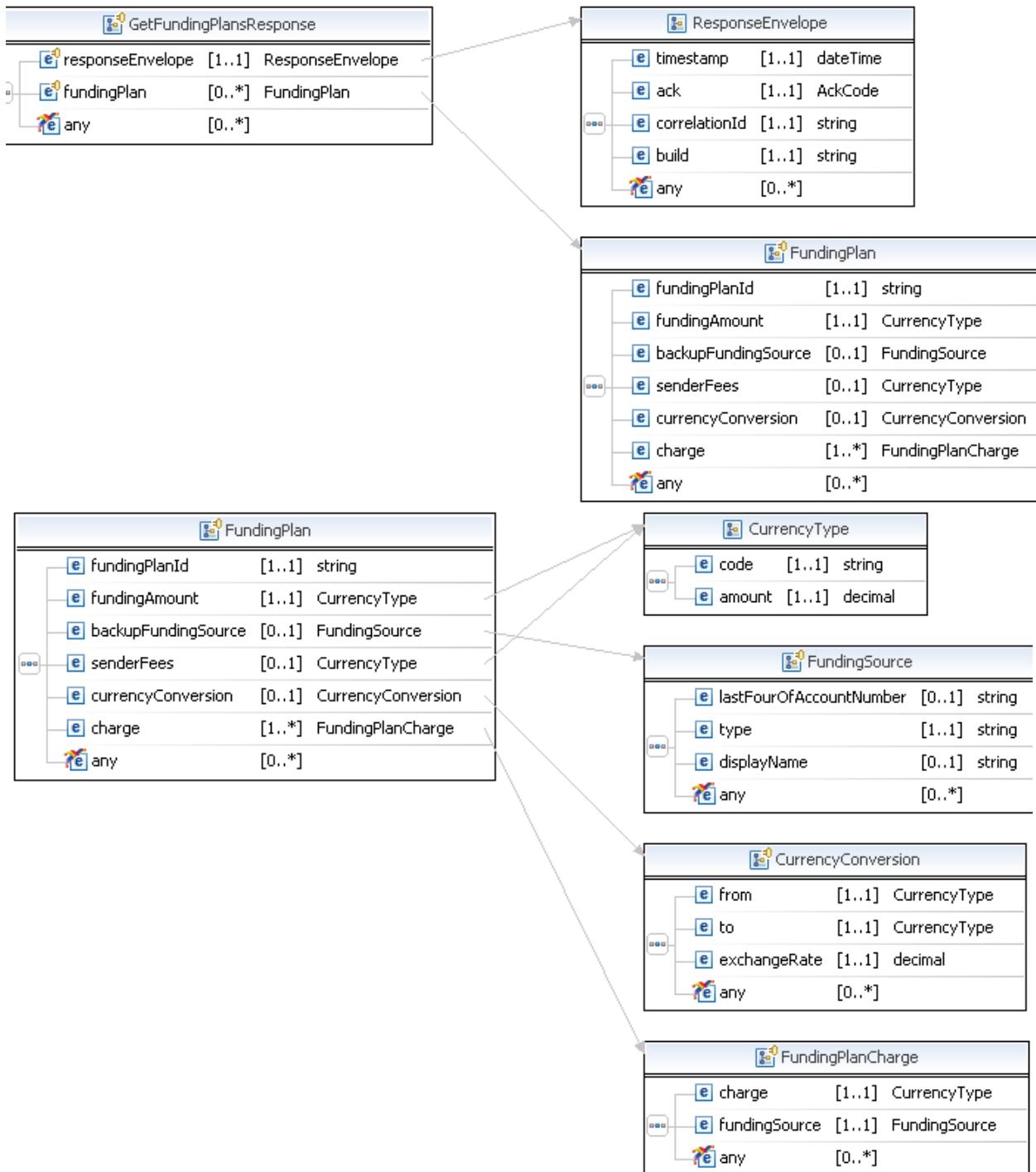
RequestEnvelope Fields

Field	Description
detailLevel	<code>common:DetailLevelCode</code> <i>(Optional)</i> The level of detail required by the client application for component such as Item, Transaction. Possible values are: <ul style="list-style-type: none">• ReturnAll – This value provides the maximum level of detail (default)
errorLanguage	<code>xs:string</code> <i>(Required)</i> The RFC 3066 language in which error messages are returned; by default it is <code>en_US</code> , which is the only language currently supported

GetFundingPlansResponse Message

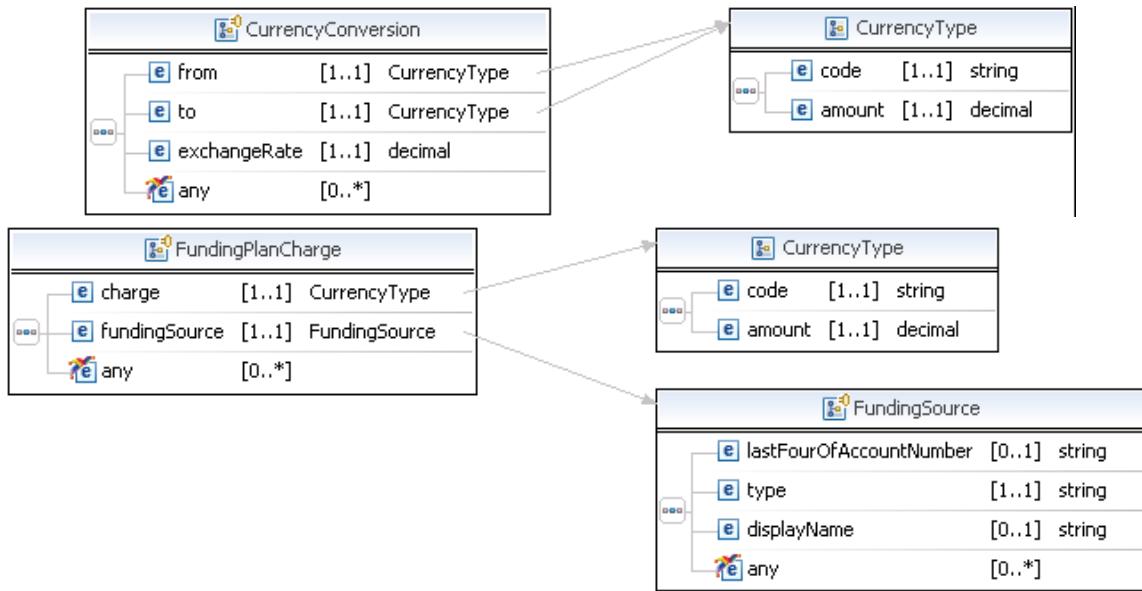
GetFundingPlans API Operation

GetFundingPlansResponse Message



GetFundingPlans API Operation

GetFundingPlansResponse Message



GetFundingPlansResponse Fields

Field	Description
FundingPlan	ap:FundingPlan One or more available funding sources for this payment.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

FundingPlan Fields

Field	Description
fundingPlanId	xs:string ID for this funding plan.
fundingAmount	common:CurrencyType Amount of available funding.
backupFundingSource	ap:FundingSource Backup funding source.
senderFees	common:CurrencyType Fees to be paid by the sender.

Field	Description
currencyConversion	ap:CurrencyConversion The currency conversion associated with the funding plan.
charge	ap:FundingPlanCharge The amount to be charged to this funding plan.

CurrencyType Fields

Field	Description
amount	xs:decimal The converted amount.
code	xs:string The currency code for the converted amount. Possible values are: <ul style="list-style-type: none"> • Australian Dollar – AUD • Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none"> • Canadian Dollar – CAD • Czech Koruna – CZK • Danish Krone – DKK • Euro – EUR • Hong Kong Dollar – HKD • Hungarian Forint – HUF • Israeli New Sheqel – ILS • Japanese Yen – JPY • Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none"> • Mexican Peso – MXN • Norwegian Krone – NOK • New Zealand Dollar – NZD • Philippine Peso – PHP • Polish Zloty – PLN • Pound Sterling – GBP • Singapore Dollar – SGD • Swedish Krona – SEK • Swiss Franc – CHF • Taiwan New Dollar – TWD • Thai Baht – THB • U.S. Dollar – USD

GetFundingPlans API Operation

GetFundingPlansResponse Message

FundingSource Fields

Field	Description
lastFourOfAccountNumber	xs:string Last 4 digits or characters in account number.
type	xs:string Type of funding source, which is one of the following values: <ul style="list-style-type: none">● UNDEFINED● BALANCE● BANK_INSTANT● BANK_DELAYED● CREDITCARD● DEBITCARD● ACCOUNTS_RECEIVABLE
displayName	ap:FundingSource Display name of funding source.

CurrencyConversion Fields

Field	Description
from	ap:CurrencyType The currency to be converted.
to	ap:CurrencyType The currency resulting from the conversion.
exchangeRate	xs:decimal The exchange rate for the <code>from</code> currency to the <code>to</code> currency.

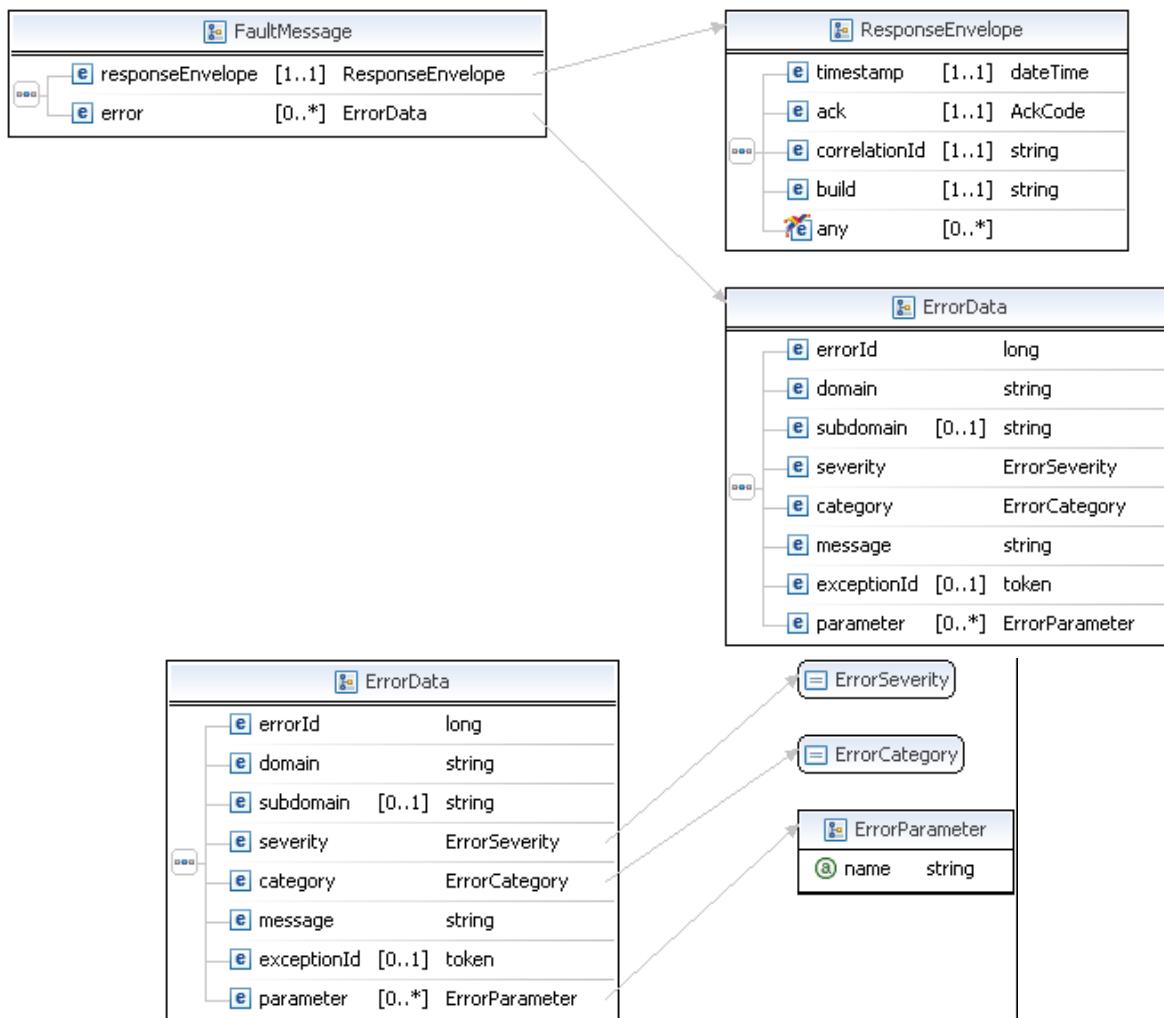
ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">● Success – Operation completed successfully● Failure – Operation failed● Warning – Warning● SuccessWithWarning – Operation completed successfully; however, there is a warning message● FailureWithWarning – Operation failed with a warning message

Field	Description
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.



GetFundingPlans API Operation

PPFault Message

FaultMessage Fields

Field	Description
error	common:ErrorData Detailed error information.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	common:ErrorCategory The location where the error occurred. Possible values are: <ul style="list-style-type: none">• System – The system encountered errors; try again• Application – The application encountered errors; try again• Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	common:ErrorParameter Represents contextual information about the error.
severity	common:ErrorSeverity The severity of the error encountered. Possible values are: <ul style="list-style-type: none">• Error – Processing of the request was interrupted• Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

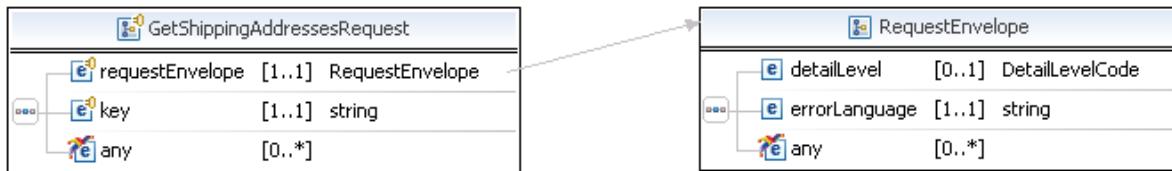
Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationID	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

GetFundingPlan Errors

Code	Message	Additional Information
520002	Internal error	
520003	Authentication failed. API credentials are incorrect.	Since 1.6.0
540031	Operation is not permitted because the credentials do not match those of the initial operation	Since 1.6.0
550001	<ul style="list-style-type: none"> • User is not allowed to perform this action. • You are not allowed to get the funding plans for this key • You do not have permission to get these funding plan details 	Since 1.6.0
580001	Invalid request: <value>	Since 1.6.0
580022	<ul style="list-style-type: none"> • Invalid request parameter: <value> • Invalid request parameter: payKey with value <value> 	Since 1.6.0
580027	This parameter is not supported	Since 1.6.0
580029	A pay key, transaction ID, or tracking ID is missing	Since 1.6.0

Use the GetShippingAddresses API operation to obtain the selected shipping address. You must have created the payment or preapproval key that identifies the account holder whose shipping address you want to obtain, or be the primary receiver of the payment or one of the parallel receivers of the payment.

GetShippingAddressesRequest Message



GetShippingAddressesRequest Fields

You must specify a value for the key field, which identifies either a payment or a preapproval.

Field	Description
key	<code>xs:string</code> <i>(Required)</i> A payment or preapproval key that identifies the account holder for whom you want to obtain the selected shipping address.
requestEnvelope	<code>common:requestEnvelope</code> <i>(Required)</i> Information common to each API operation, such as the language in which an error message is returned.

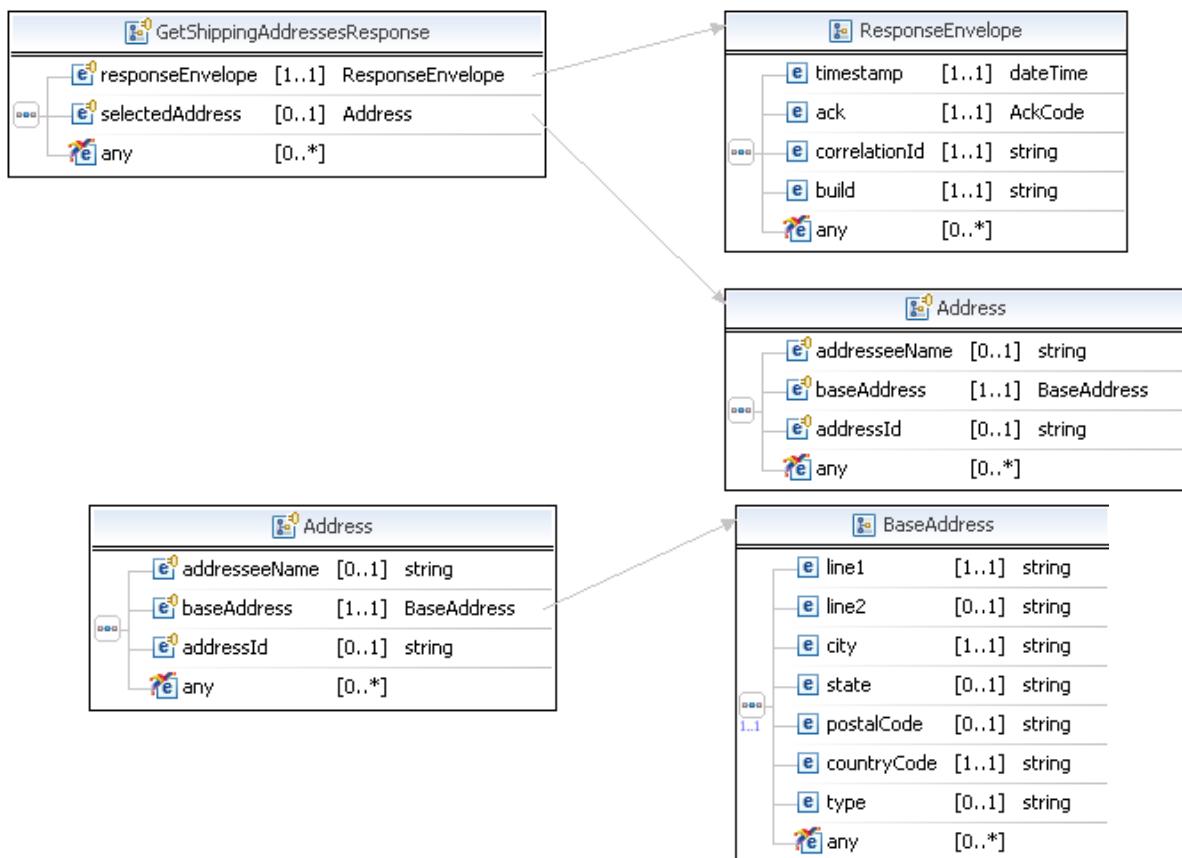
GetShippingAddresses API Operation

GetShippingAddressesResponse Message

RequestEnvelope Fields

Field	Description
detailLevel	common:DetailLevelCode <i>(Optional)</i> The level of detail required by the client application for component such as Item, Transaction. Possible values are: <ul style="list-style-type: none">• ReturnAll – This value provides the maximum level of detail (default)
errorLanguage	xs:string <i>(Required)</i> The RFC 3066 language in which error messages are returned; by default it is en_US, which is the only language currently supported

GetShippingAddressesResponse Message



GetShippingAddressesResponse Fields

Field	Description
availableAddress	ap:Address Any number of available shipping addresses.
responseEnvelope	common:ResponseEnvelope Common response information, including a timestamp and the response acknowledgement status.

Address Fields

Field	Description
addresseeName	xs:string The name associated with the address.
baseAddress	common:BaseAddress Street address.
addressId	xs:string The ID associated with the address.

BaseAddress Fields

Field	Description
city	xs:string The city of the address.
countryCode	xs:string The country code of the address.
line1	xs:string The first line of the address.
line2	xs:string An second line of the street address.
postalCode	xs:string The postal code of the address.
state	xs:string The state for the address
type	xs:string. The type of address.

GetShippingAddresses API Operation

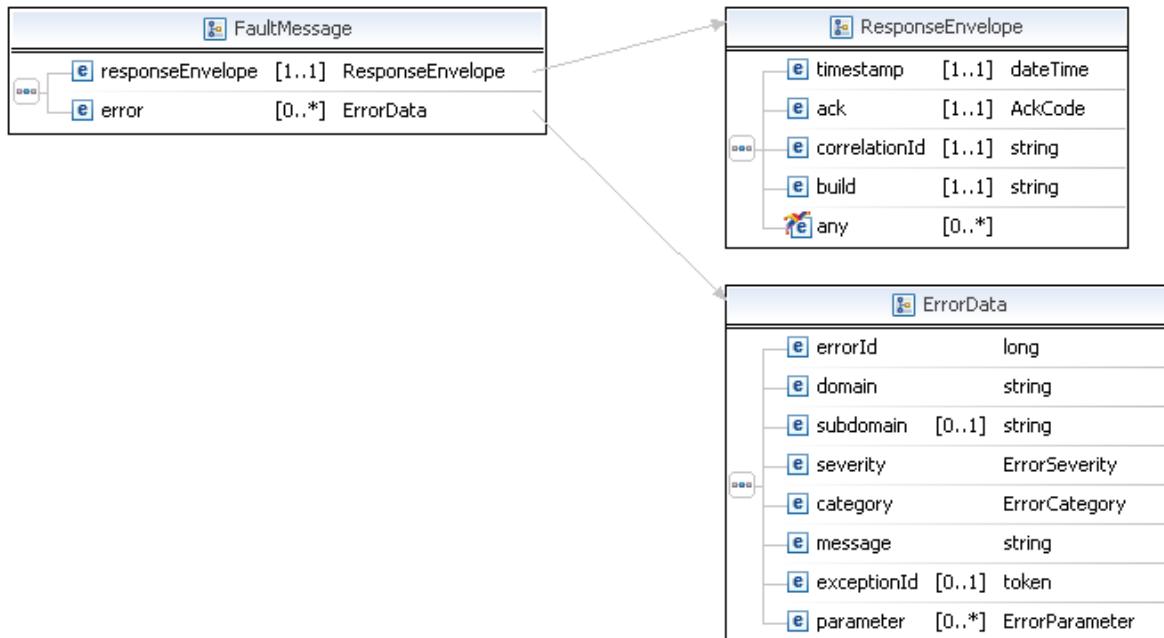
PPFault Message

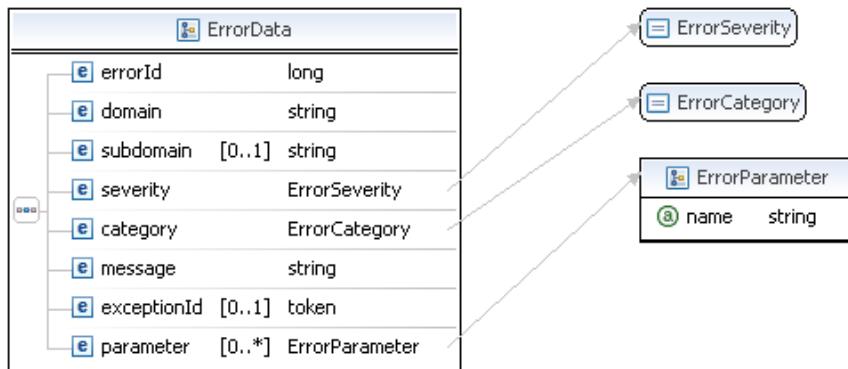
ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

PPFault Message

The PPFaultMessage returns ErrorData and the ResponseEnvelope information to your application if an error occurs.





FaultMessage Fields

Field	Description
error	<code>common:ErrorData</code> Detailed error information.
responseEnvelope	<code>common:ResponseEnvelope</code> Common response information, including a timestamp and the response acknowledgement status.

ErrorData Fields

Field	Description
category	<code>common:ErrorCategory</code> The location where the error occurred. Possible values are: <ul style="list-style-type: none"> • System – The system encountered errors; try again • Application – The application encountered errors; try again • Request – The request was incorrect
domain	The domain to which this service belongs.
errorId	A number that uniquely identifies a particular error.
exceptionID	This field is not used.
message	A description of the error.
parameter	<code>common:ErrorParameter</code> Represents contextual information about the error.

GetShippingAddresses API Operation

GetShippingAddresses Errors

Field	Description
severity	common:ErrorSeverity The severity of the error encountered. Possible values are: <ul style="list-style-type: none">• Error – Processing of the request was interrupted• Warning – Processing of the request was completed
subdomain	This field is not used.

ResponseEnvelope Fields

Field	Description
ack	common:AckCode Acknowledgement code. Possible values are: <ul style="list-style-type: none">• Success – Operation completed successfully• Failure – Operation failed• Warning – Warning• SuccessWithWarning – Operation completed successfully; however, there is a warning message• FailureWithWarning – Operation failed with a warning message
build	Build number; used only by Developer Technical Support.
correlationId	Correlation ID; used only by Developer Technical Support.
timestamp	The date on which the response was sent. The time is currently not supported.

GetShippingAddresses Errors

Code	Message	Additional Information
520002	Internal error	
520003	Authentication failed. API credentials are incorrect.	Since 1.6.0
540031	Operation is not permitted because the credentials do not match those of the initial operation	Since 1.6.0
550001	<ul style="list-style-type: none">• User is not allowed to perform this action.• You are not allowed to get the addresses for this key	Since 1.6.0
580001	Invalid request: <value>	Since 1.6.0
580022	<ul style="list-style-type: none">• Invalid request parameter: <value>• Invalid request parameter: payKey with value <value>• No sender is associated with this key	Since 1.6.0

Code	Message	Additional Information
580029	A pay key, transaction ID, or tracking ID is missing	Since 1.6.0

The web flows associated with Adaptive Payments may require you to redirect a sender's browser to PayPal. When you redirect to PayPal you must send a command and associated parameters.

Command	Description
_ap-payment	Invokes PayPal login to approve a payment using Adaptive Payments
_ap-preapproval	Invokes PayPal login to set up a preapproval using Adaptive Payments

[_ap-payment Command](#)

The sender can log in to paypal.com and approve a payment when you redirect the sender's browser to paypal.com with the _ap-payment command.

Redirect to PayPal using the _ap-payment command

https://www.paypal.com/webscr?cmd=_ap-payment&paykey=value

Parameter	Description
cmd	(Required) _ap-payment
paykey	(Required) The payment key from the Pay response message; for example, AP-12345678901234567.

Response to _ap-payment command from PayPal

PayPal redirects to the return URL you set in the Pay request.

NOTE: Any additional parameters you specify in the URL are not returned.

_ap-preapproval Command

The sender can log in to paypal.com and approve a preapproval request when you redirect the sender's browser to PayPal using the _ap-preapproval command.

Redirect to PayPal using the _ap-preapproval command

`https://www.paypal.com/webscr?cmd=_ap-preapproval&preapprovalkey=value`

Parameter	Description
cmd	(Required) _ap-preapproval
preapprovalkey	(Required) The preapproval key from the Preapproval response message; for example, PA-12345678901234567.

Response to _ap-preapproval command from PayPal

PayPal redirects to the return URL you set in the Preapproval request.

NOTE: Any additional parameters you specify in the URL are not returned.

Embedded Payment Flow Command

The sender of a payment can log in to paypal.com and approve a payment using the embedded payment flow. Your web pages must use PayPal-provided JavaScript to launch the flow.

Redirect to PayPal

`https://paypal.com/webapps/adaptivepayment/flow/pay?
paykey=value&preapprovalkey=value`

Parameter	Description
paykey	(Required) The payment key from the Pay response message; for example, AP-12345678901234567.
preapprovalkey	(Optional) The preapproval key from the Preapproval response message, which enables the Preapprove future payments option; for example, PA-12345678901234567.

Response from PayPal

PayPal redirects to the return URL you set in the Pay request.

NOTE: Any additional parameters you specify in the URL are not returned.

JavaScript Functions for Embedded Payments

You must include `https://www.paypalobjects.com/js/external/dg.js` on any page that invokes or terminates the embedded payment flow. The JavaScript functions in `dg.js` set up and control the PayPal login and payment experience.

Function and Signature	Description
<code>PAYPAL.apps.DGFlow = function (trigger: null, expType: null)</code>	<p>Sets up the embedded payment flow, where <code>trigger</code> is one or more HTML ids of events that trigger the flow and <code>expType</code> is one of the following experience type requests:</p> <ul style="list-style-type: none"> • <code>light</code> requests that the embedded payment flow displays in a lightbox (default) • <code>mini</code> requests that the embedded payment flow displays in a minibrowser <p>PayPal handles other values, including misspellings, as <code>light</code>. Opening a minibrowser in a lightbox causes errors during the flow.</p> <p>NOTE: PayPal does not honor the lightbox experience request if the sender is required to log into PayPal.</p>
<code>setTrigger: function (id)</code>	Add a trigger for the flow, where <code>id</code> is the HTML id of the event that triggers the flow. The flow must first be set up by calling <code>PAYPAL.apps.DGFlow</code> .
<code>startFlow: function (url)</code>	Explicitly start the embedded payment flow, where <code>url</code> is the URL that triggers the flow.
	<p>NOTE: You are only required to explicitly start the flow when there is no triggering event; for example, when you want to associate the embedded payment flow with playing a Flash movie.</p>
<code>closeFlow: function ()</code>	Close the minibrowser or lightbox associated with the flow. Use this function after the sender completes or cancels the payment.
	<p>NOTE: In addition to calling this function, you must explicitly close the PayPal window using the <code>close()</code> JavaScript function.</p>
<code>isOpen: function ()</code>	Determine whether the minibrowser or lightbox is still open.

PayPal sends an IPN message in response to a payment, a refund, creation of a preapproval, and cancellation of a preapproval. Your IPN message handler must be set up in the way described in the *Instant Payment Notification Guide*.

Specifically, it must guard against spoofing and it must be able to handle duplicate messages, which can occur, for example, if your application is both the caller of the Pay API and a receiver of the payment.

IMPORTANT: IPN messages for Adaptive Payments do not use the IPN message handler that you can specify in the Profile. IPN messages are only sent to the notification URL specified in the Adaptive Payment API operation's request.

Pay Message Variables

PayPal generates an IPN message that contains information about the pay request or payment in response to the Adaptive Payments Pay and ExecutePayment API operations.

Variable	Description
transaction_type	The type of transaction. Possible values are: <ul style="list-style-type: none">• Adaptive Payment PAY This notification occurs when a payment is made due to a PayRequest. The variables for the Adaptive Payment Pay notification are similar to the PaymentDetailsResponse fields.• Adjustment This can be for a chargeback, reversal, or refund; check the reason_code to see which it is.
status	The status of the payment. Possible values are: <ul style="list-style-type: none">• CREATED – The payment request was received; funds will be transferred once the payment is approved• COMPLETED – The payment was successful• INCOMPLETE – Some transfers succeeded and some failed for a parallel payment or, for a delayed chained payment, secondary receivers have not been paid• ERROR – The payment failed and all attempted transfers failed or all completed transfers were successfully reversed• REVERSALERROR – One or more transfers failed when attempting to reverse a payment• PROCESSING – The payment is in progress• PENDING – The payment is awaiting processing

Instant Payment Notifications

Pay Message Variables

Variable	Description
sender_email	Sender's email address.
action_type	Whether the Pay API is used with or without the SetPaymentOptions and ExecutePayment API operations. Possible values are: <ul style="list-style-type: none">• PAY – If you are not using the SetPaymentOptions and ExecutePayment API operations• CREATE – If you are using the SetPaymentOptions and ExecutePayment API operations
payment_request_date	The date on which the payment request was initiated.
reverse_all_parallel_payments_on_error	Whether the payment request specified to reverse parallel payments if an error occurs. Possible values are: <ul style="list-style-type: none">• true – Each parallel payment is reversed if an error occurs• false – Only incomplete payments are reversed (default)
transaction[n].id	The transaction ID, where [n] is a number from 0 to 5. For simple, single receiver payments, this number will be 0. Numbers larger than 0 indicate the payment to a particular receiver in chained and parallel payments.
transaction[n].status	The transaction status, where [n] is a number from 0 to 5. For simple single-receiver payments, this number will be 0. Numbers larger than 0 indicate the payment to a particular receiver in chained and parallel payments. Possible values are: <ul style="list-style-type: none">• Completed• Pending• Refunded
transaction[n].id_for_sender	The transaction ID for the sender, where [n] is a number from 0 to 5. For simple, single receiver payments, this number will be 0. Numbers larger than 0 indicate the payment to a particular receiver in chained and parallel payments.
transaction[n].status_for_sender_txn	The transaction status, where [n] is a number from 0 to 5. For simple single-receiver payments, this number will be 0. Numbers larger than 0 indicate the payment to a particular receiver in chained and parallel payments. Possible values are: <ul style="list-style-type: none">• SUCCESS – The sender's transaction has completed• PENDING – The transaction is awaiting further processing• CREATED – The payment request was received; funds will be transferred once approval is received• PARTIALLY_REFUNDED – Transaction was partially refunded• DENIED – The transaction was rejected by the receiver• PROCESSING – The transaction is in progress• REVERSED – The payment was returned to the sender• REFUNDED – The payment was refunded• FAILED – The payment failed
transaction[n].refund_id	The identification number for the refund

Variable	Description
transaction[n].refund_amount	The amount that was refunded.
transaction[n].refund_acount_charged	The email address of the debit account of the refund.
transaction[n].receiver	The receiver's email address for the transaction
transaction[n].invoiceId	The invoice number for this transaction
transaction[n].amount	The payment amount of the transaction
transaction[n].is_primary_receiver	Whether there is a primary receiver for this transaction, which indicates whether the transaction is a chained payment. Possible values are: <ul style="list-style-type: none">• true – There is a primary receiver (chained payment)• false – There is no primary receiver (simple or parallel payment)
return_url	The URL to which the sender's browser is redirected after approving a payment on paypal.com. Use the pay key to identify the payment as follows: payKey=\${payKey}.
cancel_url	The URL to which the sender's browser is redirected if the sender cancels the approval for a payment on paypal.com. Use the pay key to identify the payment as follows: payKey=\${payKey}.
ipn_notification_url	The URL to which all IPN messages for this payment are sent.
pay_key	The pay key that identifies this payment. This is a token that is assigned by the Pay API after a PayRequest message is received and can be used in other Adaptive Payments APIs as well as the cancelURL and returnUrl to identify this payment. The pay key is valid for 3 hours.
memo	A note associated with the payment.
fees_payer	The payer of PayPal fees. Possible values are: <ul style="list-style-type: none">• SENDER – Sender pays all fees (for personal, implicit simple/parallel payments; do not use for chained or unilateral payments)• PRIMARYRECEIVER – Primary receiver pays all fees (chained payments only)• EACHRECEIVER – Each receiver pays their own fee (default, personal and unilateral payments)• SECONDARYONLY – Secondary receivers pay all fees (use only for chained payments with one secondary receiver)
trackingId	The tracking ID that was specified for this payment in the PaymentDetailsRequest message.
preapproval_key	The preapproval key returned after a PreapprovalRequest, or the preapproval key that identifies the preapproval key sent with a PayRequest.

Instant Payment Notifications

Preapproval Message Variables

Variable	Description
reason_code	Whether this transaction is a chargeback, partial, or reversal. Possible values are: <ul style="list-style-type: none">• Chargeback Settlement – Transaction is a chargeback• Admin reversal – Transaction was reversed by PayPal administrators• Refund – Transaction was partially or fully refunded

Preapproval Message Variables

PayPal generates an IPN message that contains information about a preapproval in response to the Adaptive Payments Preapproval API operation.

Variable	Description
transaction_type	The type of transaction. For a preapproval, this variable returns Adaptive Payment Preapproval. NOTE: If this variable is set to Adaptive Payment Pay or Adjustment, refer to the Pay Message Variable section.
preapproval_key	The preapproval key returned after a PreapprovalRequest.
approved	Whether the preapproval request was approved. Possible values are: <ul style="list-style-type: none">• true – The preapproval was approved• false – The preapproval was denied
cancel_url	The URL to which the sender's browser is redirected if the sender decides to cancel the preapproval as requested. Use the preapproval key to identify the payment as follows: preapprovalKey=\${preapprovalKey}
current_number_of_payments	The current number of payments made for this preapproval.
current_total_amount_of_all_payments	The current total of payments made for this preapproval.
current_period_attempts	The current number of attempts this period for this preapproval.

Variable	Description
currencyCode	<p>The currency code. Possible values are:</p> <ul style="list-style-type: none"> • Australian Dollar – AUD • Brazilian Real – BRL <p>NOTE: The Real is supported as a payment currency and currency balance only for Brazilian PayPal accounts.</p> <ul style="list-style-type: none"> • Canadian Dollar – CAD • Czech Koruna – CZK • Danish Krone – DKK • Euro – EUR • Hong Kong Dollar – HKD • Hungarian Forint – HUF • Israeli New Sheqel – ILS • Japanese Yen – JPY • Malaysian Ringgit – MYR <p>NOTE: The Ringgit is supported as a payment currency and currency balance only for Malaysian PayPal accounts.</p> <ul style="list-style-type: none"> • Mexican Peso – MXN • Norwegian Krone – NOK • New Zealand Dollar – NZD • Philippine Peso – PHP • Polish Zloty – PLN • Pound Sterling – GBP • Singapore Dollar – SGD • Swedish Krona – SEK • Swiss Franc – CHF • Taiwan New Dollar – TWD • Thai Baht – THB • U.S. Dollar – USD
date_of_month	The day of the month on which a monthly payment is to be made. A number between 1 and 31 indicates the day of the month. A value of 0 indicates that the payment can be made on any day.
day_of_week	The day of the week that a weekly payment is to be made. Possible values are:
	<ul style="list-style-type: none"> • NO_DAY_SPECIFIED • SUNDAY • MONDAY • TUESDAY • WEDNESDAY • THURSDAY • FRIDAY • SATURDAY
starting_date	First date for which the preapproval is valid.
ending_date	Last date for which the preapproval is valid. Time is currently not supported.

Instant Payment Notifications

Preapproval Message Variables

Variable	Description
max_total_amount_of_all_payments	The preapproved maximum total amount of all payments.
max_amount_per_payment	The preapproved maximum amount of all payments.
max_number_of_payments	The maximum number of payments that is preapproved.
payment_period	<p>The payment period. Possible values are:</p> <ul style="list-style-type: none">● NO_PERIOD_SPECIFIED● DAILY● WEEKLY● BIWEEKLY● SEMIMONTHLY● MONTHLY● ANNUALLY
pin_type	Whether a personal identification number (PIN) is required. It is one of the following values: <ul style="list-style-type: none">● NOT_REQUIRED – A PIN is not required● REQUIRED – A PIN is required
sender_email	The sender's email address.

Older versions of the Adaptive Payments API can still be used; however, PayPal recommends using the latest version whenever possible. Unless specifically specified, API operations default to the latest version.

1.5.0 Features

Adaptive Payments version 1.5.0 provides the new features for the APIs, such as the ability to delay a chained payment and to obtain the reason a payment is pending.

In addition to API changes, version 1.5.0 also offers new features that do not affect the APIs, such as guest payments, and ELV support.

Ability to Delay a Chained Payment Feature in Version 1.6.0

You can now set up chained payments to process the transaction leg to the primary receiver, and delay the payment to the secondary receivers. To do this, you use the Pay or ExecutePayment call and pass the new parameter `actionType` with `PAY_PRIMARY`.

PaymentDetails Pending Reason Feature in Version 1.6.0

The `PaymentDetailsResponse` now returns the new `pendingReason` parameter. This parameter returns a string describing the reason why a transaction is pending. Possible values are:

- `ADDRESS_CONFIRMATION` – The payment is pending because the sender does not have a confirmed address on file and receiver's Payment Receiving Preferences is set for manually accepting payments or denying each of these payments.
- `ECHECK` – The payment is pending because it was made by an eCheck that has not yet cleared.
- `INTERNATIONAL` – The payment is pending because the receiver holds a non-U.S. account and does not have a withdrawal mechanism. The receiver must manually accept or deny this payment from the Account Overview.
- `MULTI_CURRENCY` – The receiver does not have a balance in the currency sent, and does not have the Payment Receiving Preferences set to automatically convert and accept this payment. Receiver must manually accept or deny this payment from the Account Overview.

1.4.0 Features

- **UNILATERAL** – The payment is pending because it was made to an email address that is not yet registered or confirmed.
- **UPGRADE** – The payment is pending because it was made via credit card and the receiver must upgrade the account to a Business account or Premier status to receive the funds. It can also mean that receiver has reached the monthly limit for transactions on the account.
- **VERIFY** – The payment is pending because the receiver is not yet verified.
- **RISK** – The payment is pending while it is being reviewed by PayPal for risk.
- **OTHER** – The payment is pending for a reason other than those listed above. For more information, contact PayPal Customer Service.

1.4.0 Features

Release 1.4.0 provides the new API operations: `ExecutePayment`, `GetPaymentOptions`, and `SetPaymentOptions`. The new API operations work with the `actiontype CREATE` in the `Pay` API operation.

New API Operations for Version 1.4.0

New API Operation	Description
<code>ExecutePayment</code>	Execute a payment created with a <code>PayRequest</code> that had the <code>actionType</code> set to <code>CREATE</code> .
<code>GetPaymentOptions</code>	Retrieve payment settings specified with the <code>SetPaymentOptions</code> API operation.
<code>SetPaymentOptions</code>	Set payment options to be executed with the <code>ExecutePayment</code> API operation.

Changes to the Pay API Operation

Changes to PayRequest and PayResponse Message

Field	Description
actionType	<p>New CREATE value:</p> <ul style="list-style-type: none"> • PAY – Use this option if you are not using the Pay request in combinations with the ExecutePayment request. • CREATE – Use this option to set up the payment instructions with the Pay request and then execute the payment at a later time with the ExecutePayment request.
paymentType	<p>New person-to-person values PERSONAL and CASHADVANCE.</p> <ul style="list-style-type: none"> • •
Receiver.PhoneNumberType	<p>New field: common : PhoneNumberType Allows you to pass a receiver's mobile phone number, including the country code and extension.</p> <p>NOTE: This feature is available for applications with special permission level.</p>

Changes to the PaymentDetails API Operation

Changes to $\frac{3}{4}$ messages

Field	Description
actionType	<p>New CREATE value:</p> <ul style="list-style-type: none"> • PAY – Use this option if you are not using the Pay request in combinations with the ExecutePayment request. • CREATE – Use this option to set up the payment instructions with the Pay request and then execute the payment at a later time with the ExecutePayment request.

Older Versions of the Adaptive Payments API

1.4.0 Features

Field	Description
paymentType	<p>New person-to-person values PERSONAL and CASHADVANCE.</p> <p>Allowable values are:</p> <ul style="list-style-type: none">• GOODS – This is a payment for goods• SERVICE – This is a payment for services (default)• PERSONAL – This is a person-to-person payment• CASHADVANCE – This is a person-to-person payment for a cash advance <p>Person-to-person payments can only be made under the following conditions:</p> <ul style="list-style-type: none">• simple and parallel payments (not chained)• all transactions within the single Pay request must be person-to-person• feesPayer field set to EACHRECEIVER or SENDER.• this feature is available for applications with special permission level
Receiver.PhoneNumberType	<p>New field: common:PhoneNumberType</p> <p>Allows you to pass a receiver's mobile phone number, including the country code and extension.</p> <p>NOTE: This feature is available for applications with special permission level.</p>

Changes to the PreapprovalDetails API Operation

Changes to PreapprovalDetailsRequest Message

Field	Description
getBillingAddress	<p>New field: xs:boolean</p> <p>(Optional) An option that lets you retrieve a list of billing addresses for the sender.</p> <p>Allowable values are:</p> <ul style="list-style-type: none"> • true – Includes the billing address in the response • false – Omits the billing address from the response (default) <p>NOTE: This field is available only to API callers with advanced permission levels. For information, refer to the section Adaptive Payments Permission Levels.</p>

Changes to the PreapprovalDetails API Operation

Changes to PreapprovalDetailsResponse Message

Field	Description
addresslist	<p>New field: ap:AddressList</p> <p>Returns a list of billing addresses.</p>
AddressList	<p>New field: ap:Address</p>
Address.addresseeName	<p>xs:string</p> <p>The name of the person that is billed.</p>
Address.baseAddress	<p>New field: common:BaseAddress</p>
BaseAddress.city	<p>xs:string</p> <p>The city of the billing address</p>
BaseAddress.countryCode	<p>xs:string</p> <p>The country code for the billing address</p>
BaseAddress.line1	<p>xs:string</p> <p>The streetname for the billing address</p>
BaseAddress.line2	<p>xs:string</p> <p>An extra line for the street address</p>
BaseAddress.postalCode	<p>xs:string</p> <p>The postal code of the billing address</p>

Older Versions of the Adaptive Payments API

1.3.0 Features

Field	Description
BaseAddress.state	<code>xs:string</code> The state for the billing address
BaseAddress.type	<code>xs:string</code> The type of billing address. Possible values are: <ul style="list-style-type: none">• BILLING – This is a billing address

Changes to the Refund API Operation

Changes to RefundRequest and RefundResponse Message

Field	Description
Receiver.PhoneNumberType	New field: <code>common:PhoneNumberType</code> Allows you to pass a receiver's phone number, including the country code and extension. NOTE: This feature is available for applications with special permission level.

1.3.0 Features

Version 1.3.0 includes changes to existing APIs as well new APIs: ConvertCurrency, CancelPreapproval. It also includes the new PPFaultMessage.

Overview of Changes for Version 1.3.0

These are the main changes for version 1.3.0:

- **Consumer signups for new PayPal accounts**

Consumers can now sign up for a PayPal account during the payment preapproval web flow. Consumers can also sign up for a PayPal account during a payment web flow and then complete the payment in the same session.

- **Preapproval cancellations**

This release includes a new API operation to allow consumers to cancel preapprovals: the CancelPreapproval API operation. In past releases, consumers could only cancel preapprovals through paypal.com. For more information, refer to the CancelPreapproval API Operation section.

- **Getting Foreign Exchange rates for a list of payment amounts**

This release provides a new API Operation, ConvertCurrency, that lets you convert a given amount in one currency to an equivalent amount in another currency based on the

current Foreign Exchange rate. For more information, refer to the ConvertCurrency API Operation section.

- **Payments to unregistered PayPal email addresses**

With the Adaptive Payments Pay API operation, you can make payments to a person who does not have a PayPal account. In essence, you can send a payment request that includes an email address for a receiver in which the receiver's email address is not linked to a registered PayPal account. PayPal notifies the receiver at this email address to create an account and claim the payment. Supporting features include email notifications and IPN messages to the sender, receiver, and API caller, plus the ability to cancel unclaimed payments.

HTTP Header Changes

In previous versions, the application ID was passed in the ClientDetails element `applicationId`. Version 1.3.0 requires a new header to be passed to identify the application ID. The `ClientDetails.applicationId` element is no longer necessary.

Changes to the Pay API Operation for Version 1.3.0

Changes to PayRequest Message

Field	Description
ClientDetails.applicationId	This field is no longer required. The application ID is passed in a new HTTP header. For information, refer to the Specifying Application and Device Information section.
ClientDetails.ipAddress	This field is no longer required.
fundingConstraint	<p>New field: <code>ap:FundingConstraint</code></p> <p>Specifies a list of allowed funding types for the payment. This list can be in any order. If this field is omitted, the payment can be funded by any funding type that is supported by Adaptive Payments.</p> <p>NOTE: This feature is available for applications with special permission level.</p>
FundingTypeList.fundingTypeInfo	<p>New field: <code>ap:FundingTypeInfo</code></p> <p>Specifies a list of allowed funding types for the payment. This list can be in any order. If this field is omitted, the payment can be funded by any funding type that is supported by Adaptive Payments.</p> <p>NOTE: This feature is available for applications with special permission level.</p>
fundingType	<p>New field: <code>xs:string</code></p> <p>Specifies the allowed funding types that can be used as funding sources for the payment.</p> <p>Allowable values are:</p> <ul style="list-style-type: none"> • ECHECK – Electronic check • BALANCE – PayPal account balance • CREDITCARD – Credit card <p>NOTE: This feature is available for applications with special permission level.</p>
logdefaultShippingAddress	This field has been removed. Pay requests that use this field are executed, but the logdefaultShippingAddress is ignored.
Receiver.paymentType	<p>New field: <code>xs:string</code></p> <p>The transaction subtype for the payment. Allowable values are:</p> <ul style="list-style-type: none"> • GOODS – This is a payment for goods • SERVICE – This is a payment for services (default)

Field	Description
Receiver.email	<code>xs:string</code> In this release, the receiver's email address does not need to be registered with paypal.com. If so, the receiver cannot claim the payment until a PayPal account has been created for the email address.
senderEmail	<code>xs:string</code> In previous releases, this field required the email address to be registered with paypal.com. In the 1.3.0 release, you have the option to omit the field from the payment request. The consumer can either log in using an existing paypal account (that is not associated with a receiver) or create a new account during the payment flow.

Changes to the PaymentDetails API Operation for Version 1.3.0

Changes to PaymentDetailsResponse messages

Field	Description
fundingConstraint	<p>New field: <code>ap:FundingConstraint</code></p> <p>Specifies a list of allowed funding types for the payment. This list can be in any order. If this field is omitted, the payment can be funded by any funding type that is supported by Adaptive Payments.</p> <p>NOTE: This feature is available for applications with a special permission level.</p>
FundingTypeList.fundingTypeInfo	<p>New field: <code>ap:FundingTypeInfo</code></p> <p>Specifies a list of allowed funding types for the payment. This list can be in any order. If this field is omitted, the payment can be funded by any funding type that is supported by Adaptive Payments.</p> <p>NOTE: This feature is available for applications with a special permission level.</p>
fundingType	<p>New field: <code>xs:string</code></p> <p>Specifies the allowed funding type that can be used as the funding source for the payment.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> • ECHECK – Electronic check • BALANCE – PayPal account balance • CREDITCARD – Credit card <p>NOTE: This feature is available for applications with a special permission level.</p>
PaymentInfo.senderTransactionID	<p><code>xs:string</code></p> <p>The identification number of the sender of the payment. For previous releases, the response included only information about the receiver portion of the transaction. With 1.3.0, the response includes information about the sender portion of the transaction. This feature accommodates payments to a receiver's email address that is not linked to a registered PayPal account.</p>

Field	Description
PaymentInfo.senderTransactionStatus	xs:string The status of the sender of the payment. For previous releases, the response included only information about the receiver portion of the transaction. With 1.3.0, the response includes information about the sender portion of the transaction. This feature accommodates payments to a receiver's email address that is not linked to a registered PayPal account.

1.2.0 Features

Version 1.2.0 of the Adaptive Payments API introduces changes to several Adaptive Payments API operations. These changes include backward-compatible changes to request and response messages as well as additional error messages.

Changes to the Pay API Operation for Version 1.2.0

Changes to PayRequest messages

Field	Description
ClientDetails.deviceID	This field is now optional; it is no longer required.
Receiver.invoiceID	New field: <code>xs:string</code> <i>(Optional)</i> The invoice number for the payment

Changes to PayResponse message

Field	Description
Receiver.invoiceID	New field: <code>xs:string</code> The invoice number for the payment

Changes to the Payment Details API Operation for Version 1.2.0

Changes to PaymentDetailsReponse message

Field	Description
Receiver.invoiceID	New field: <code>xs:string</code> The invoice number for the payment

Changes to Preapproval API Operation for Version 1.2.0

Changes to PreapprovalRequest messages

Field	Description
ipnNotificationUrl	New field: <code>xs:string</code> <i>(Optional)</i> The URL to which you want all IPN messages for this preapproval to be sent. This URL supersedes the IPN notification URL in your profile.
memo	New field: <code>xs:string</code> <i>(Required)</i> A note about the preapproval.
ClientDetails.deviceID	This field is now optional; it is no longer required.

Changes to Preapproval Details API Operation for Version 1.2.0

Changes to PreapprovalDetailsResponse message

Field	Description
ipnNotificationUrl	New field: <code>xs:string</code> The URL to which all IPN messages for this preapproval are sent.
memo	New field: <code>xs:string</code> A note about the preapproval.

1.1.0 Features

Version 1.1.0 of the Adaptive Accounts API introduces changes to all Adaptive Payments API operations. These changes include backward compatible changes to request and response messages as well as additional error messages.

Older Versions of the Adaptive Payments API

1.1.0 Features

Changes to the Pay API Operation for Version 1.1.0

Changes to PayRequest message

Field	Description
logDefaultShippingAddress s	New field: <code>xs:boolean</code> <i>(Optional)</i> Whether or not to associate the receiver's default shipping address with the transaction.
clientDetails.deviceId	This field is now optional; it is no longer required.

Changes to PayResponse message

Field	Description
payErrorList	New field: <code>ap:PayErrorList</code> Information about why a payment failed.

New Pay error messages

Code	Message	Additional Information
580022	Invalid request: ~1	Invalid request parameter; for example an invalid application ID or language code
580026	The invoice id (~1) already exists for receiver ~2 and cannot be duplicated	
580027	Payment is not allowed to be pending	
580029	Invalid request: ~1	Missing request parameter

Changes to PaymentDetails API Operation for Version 1.1.0

Changes to PaymentDetailsResponse

Field	Description
logDefaultShippingAddress s	New field: <code>xs:string</code> Indicates when the Pay operation was called for this payKey.
trackingId	Empty fields are not returned; this is now enforced for trackingID.

New PaymentDetails error messages

Code	Message	Additional Information
540031	You do not have permission to get the details of this Payment	PaymentDetails

Code	Message	Additional Information
580022	Invalid request: ~1	Invalid request parameter; for example an invalid application ID or language code
580029	Invalid request: ~1	Missing request parameter

Older Versions of the Adaptive Payments API

1.1.0 Features

Changes to Preapproval API Operation for Version 1.1.0

Changes to PreapprovalRequest message

Field	Description
maxNumberOfPayments	This field is now optional; it is no longer required.
senderEmail	This field is now optional; it is no longer required. If not specified, the email address of the sender who logs in to approve the request becomes the email address associated with the preapproval key.
clientDetails.deviceId	This field is now optional; it is no longer required.

New Preapproval error messages

Code	Message	Additional Information
580022	Invalid request: ~1	Invalid request parameter; for example an invalid application ID or language code
580029	Invalid request: ~1	Missing request parameter

Changes to PreapprovalDetails API Operation for Version 1.1.0

Changes to PreapprovalDetailsResponse message

Field	Description
senderEmail	The sender's email address is only available when you explicitly set the email address in the PreapprovalRequest message.
maxNumberOfPayments	The preapproved maximum number of payments is only available when you explicitly set the email address in the PreapprovalRequest message.

New PreapprovalDetails error messages

Code	Message	Additional Information
540031	You do not have permission to get the details of this Preapproval	
580022	Invalid request: ~1	Invalid request parameter; for example an invalid application ID or language code
580029	Invalid request: ~1	Missing request parameter

Changes to Refund API Operation for Version 1.1.0

Changes to RefundResponse message

Field	Description
refundInfoList.refundInf o.errorList	New field: <code>xs:string</code> Information about why a refund failed.

New Refund error messages

Code	Message	Additional Information
540031	You do not have permission to get the details of this Refund	
580022	Invalid request: ~1	Invalid request parameter; for example an invalid application ID or language code
580029	Invalid request: ~1	Missing request parameter

Index

Symbols

_ap-payment flow command 229
_ap-preapproval flow command 230

A

ack field 87, 89, 109, 111, 120, 122, 133, 135, 143, 145, 156, 158, 168, 170, 174, 177, 192, 194, 207, 209, 216, 219, 224, 226
action_type IPN variable 234
actionType request field 70
actionType response field 102
Address response fields 167, 223
addresseeName response field 167, 223
addressId response field 167, 223
AddressList response field 167
addresslist response field 164
AddressList response fields 167
allowedFundingType request field 74
amount request field 75, 184, 200
amount response field 82, 85, 107, 119, 191, 205, 215
applicationId request field 73, 154
approved IPN variable 236
approved response field 164
availableAddress response field 223

B

backupFundingSource response field 82, 214
BaseAddress request fields 167, 223
baseAddress response field 167, 223
baseAmount response field 205
baseAmountList request field 200
build field 87, 89, 109, 111, 120, 122, 133, 135, 143, 145, 156, 158, 168, 170, 174, 177, 192, 194, 207, 209, 217, 219, 224, 226
businessName request field 140
businessName response field 130

C

cancel_url IPN variable 235, 236
CancelPreapproval API operation 173
CancelPreapprovalRequest fields 173
CancelPreapprovalRequest message 173
CancelPreapprovalResponse fields 174
CancelPreapprovalResponse message 174
cancelUrl request field 70
cancelUrl request fields 151
cancelUrl response field 102, 164
category field 85, 88, 110, 119, 122, 135, 145, 158, 169, 176, 191, 194, 208, 218, 225
charge response field 82, 84, 215
city response field 167, 223
clientDetails request field 71
ClientDetails request fields 73, 154
clientDetails request fields 151
closeFlow JavaScript function 231
code request field 201
code response field 83, 206, 215
ConversionCurrencyResponse fields 205
ConvertCurrency API operation 199
ConvertCurrencyRequest fields 200
ConvertCurrencyRequest message 199
ConvertCurrencyResponse message 204
convertToCurrencyList request field 200
correlationId field 87, 89, 109, 111, 120, 122, 133, 135, 143, 145, 156, 158, 168, 170, 174, 177, 192, 194, 207, 209, 217, 219, 224, 226
countryCode request field 76, 142, 185
countryCode response field 86, 108, 131, 133, 140, 167, 192, 223
curPayments response field 164
curPaymentsAmount response field 164
curPeriodAttempts response field 164
curPeriodEndingDate response field 164
currency request field 202
currency response field 206
currencyCode request field 71, 183, 203
currencyCode request fields 152
currencyCode response field 103, 165, 188

CurrencyCodeList request fields 203
currencyConversion response field 82, 215
CurrencyConversion response fields 84, 216
currencyConversionList response field 205
CurrencyConversionList response fields 205
CurrencyConversionTable response fields 205
CurrencyList request fields 202
currencyList response field 205
CurrencyList response fields 206
CurrencyType request fields 200
CurrencyType response fields 82, 205, 215
current_number_of_payments IPN variable 236
current_period_attempts IPN variable 236
current_total_amount_of_all_payments IPN
variable 236
customerId request field 73, 154
customerType request field 73, 155
customId request field 141
customId response field 132

D

dateOfMonth request fields 152
dateOfMonth response field 165
dayOfWeek request fields 153
dayOfWeek response field 165
defaultFundingPlan response field 81
description request field 141
description response field 132
detailLevel request field 77, 100, 116, 127, 143, 155,
162, 173, 185, 203, 212, 222
deviceId request field 73, 155
displayName response field 84, 131, 140, 216
displayOptions request field 139
DisplayOptions request fields 139
displayOptions response field 129
DisplayOptions response fields 130
domain field 85, 88, 110, 119, 122, 135, 145, 158, 169,
176, 191, 194, 208, 218, 225

E

email request field 75, 77, 142, 184
email response field 85, 108, 119, 131, 133, 140, 191
emailHeaderImageUrl request field 139
emailHeaderImageUrl response field 130
emailMarketingImageUrl request field 139

emailMarketingImageUrl response field 130
embedded payment flow command 230
encryptedRefundTransactionId response field 189
endingDate request fields 153
endingDate response field 165
error PPFault field 88, 110, 121, 134, 144, 157, 169,
176, 193, 208, 218, 225
error response field 84, 118
ErrorData fields 85, 88, 110, 119, 122, 135, 145, 158,
169, 176, 191, 194, 208, 218, 225
errorId field 85, 88, 110, 119, 122, 135, 145, 158, 169,
176, 191, 194, 208, 218, 225
errorLanguage request field 77, 100, 116, 127, 143,
155, 162, 174, 185, 204, 212, 222
errorList response field 189
estimatedAmountTable response field 205
exceptionID field 85, 88, 110, 119, 122, 135, 145, 158,
169, 176, 191, 194, 208, 218, 225
exchangeRate response field 84, 216
ExecutePayment API operation 115
ExecutePaymentRequest message 115
ExecutePaymentRequest request fields 115
ExecutePaymentResponse fields 118
ExecutePaymentResponse message 116
extension request field 76, 142, 185
extension response field 86, 108, 133, 192

F

FaultMessage PPFault fields 88, 110, 121, 134, 144,
157, 169, 176, 193, 208, 218, 225
fees_payer IPN variable 235
feesPayer request field 71
feesPayer response field 103
firstName response field 131, 140
from response field 84, 216
fundingAmount response field 82, 214
fundingConstraint request field 72
FundingConstraint request fields 74
FundingPlan fields 82, 214
FundingPlan response field 214
FundingPlanCharge response fields 84
fundingPlanId request field 115
fundingPlanId response field 82, 214
fundingSource response field 84
FundingSource response fields 83, 216
fundingType request field 75
fundingType response field 105

fundingTypeInfo request field 74
 FundingTypeInfo request fields 75
 fundingTypeInfo response field 105
 FundingTypeInfo response fields 105
 FundingTypeList request fields 74
 fundingtypeList response field 104
 FundingTypeList response fields 105

G

geoLocation request field 73, 155
 getBillingAddress request field 161
 GetFundingPlans API operation 211
 GetFundingPlansRequest fields 212
 GetFundingPlansRequest message 211
 GetFundingPlansResponse fields 214
 GetFundingPlansResponse message 212
 GetPaymentOptions API operation 127
 GetPaymentOptionsRequest fields 127
 GetPaymentOptionsRequest message 127
 GetPaymentOptionsResponse fields 129
 GetPaymentOptionsResponse message 128
 GetShippingAddresses API operation 221
 GetShippingAddressesRequest fields 221
 GetShippingAddressesRequest message 221
 GetShippingAddressesResponse fields 223
 GetShippingAddressesResponse message 222

H

headerImageUrl request field 140

I

identifier request field 142
 identifier response field 132
 initiatingEntity request field 139
 initiatingEntity response field 130
 InitiatingEntity response fields 130, 140
 Institution Customer response fields 131, 140
 institutionCustomer response field 130, 140
 institutionCustomerId response field 131, 140
 institutionId response field 131, 140
 invoiceData request field 141
 InvoiceData request fields 141
 invoiceData response field 132
 InvoiceData response fields 132

invoiceId request field 75, 184
 invoiceId response field 86, 108, 119, 191
 InvoiceItem request fields 142
 InvoiceItem response fields 132
 ipAddress request field 73, 155
 ipn_notification_url IPN variable 235
 ipnNotificationUrl request field 72
 ipnNotificationUrl request fields 153
 ipnNotificationUrl response field 104, 165
 isOpen JavaScript function 231
 item request field 141
 item response field 132

J

JavaScript functions 231

K

key request field 221

L

lastFourOfAccountNumber response field 83, 216
 lastName response field 131, 141
 line1 response field 167, 223
 line2 response field 167, 223

M

maxAmountPerPayment request fields 153
 maxAmountPerPayment response field 166
 maxNumberOfPayments request fields 153
 maxNumberOfPayments response field 166
 maxNumberOfPaymentsPerPeriod request fields 153
 maxNumberOfPaymentsPerPeriod response field 166
 maxTotalAmountOfAllPayments request fields 153
 maxTotalAmountOfAllPayments response field 166
 memo IPN variable 235
 memo request field 72
 memo request fields 153
 memo response field 104, 166
 message field 85, 88, 110, 119, 122, 135, 145, 158, 169,
 176, 191, 194, 208, 218, 225
 model request field 73, 155

N

name request field 142
name response field 132

P

parameter field 85, 88, 110, 119, 122, 135, 145, 158, 169, 176, 191, 194, 208, 218, 225
partnerName request field 74, 155
Pay API operation 65
Pay Message variables 233
Pay Receiver request fields 75
pay_key IPN variable 235
payError response field 82, 118
PayError response fields 84, 118
payErrorList response field 81, 118
PayErrorList response fields 82, 118
payKey request field 115, 127, 139, 183, 212
payKey response field 81, 104, 129
payment_request_date IPN variable 234
PaymentDetails API operation 99
PaymentDetailsRequest message 99
PaymentDetailsResponse fields 100, 102
paymentExecStatus response field 81, 118
paymentInfo response field 106
PaymentInfo response fields 106
paymentInfoList response field 104
PaymentInfoList response fields 106
paymentPeriod request fields 154
paymentPeriod response field 166
paymentSubType request field 76, 184
paymentSubType response field 86, 108, 120, 191
paymentType request field 76, 184
paymentType response field 86, 108, 120, 191
PAYPAL.apps.DGFlow JavaScript function 231
PayRequest fields 70
PayRequest message 69
PayResponse fields 81
PayResponse message 77
pendingReason response field 106
pendingRefund response field 106
phone request field 76, 77, 142, 184
phone response field 86, 108, 120, 133, 192
phoneNumber request field 76, 142, 185
PhoneNumber request fields 76, 142, 185
phoneNumber response field 86, 108, 133, 192

PhoneNumber response fields 86, 108, 133, 192
pin request field 72
pinType request fields 154
pinType response field 166
PPFault message 87, 109, 120, 134, 144, 156, 168, 175, 193, 207, 217, 224
Preapproval Message variables 236
preapproval_key IPN variable 235, 236
PreapprovalDetails API operation 161
PreapprovalDetailsRequest fields 161
PreapprovalDetailsRequest message 161
PreapprovalDetailsResponse fields 164
PreapprovalDetailsResponse message 162
preapprovalKey request field 72, 161, 173
preapprovalKey response field 104
preapprovalKey response fields 156
PreapprovalRequest fields 151
PreapprovalRequest message 150
PreapprovalResponse fields 156
price request field 142
price response field 132
primary request field 76, 185
primary response field 86, 108, 120, 192

R

reason_code IPN variable 236
receiver request field 75, 141, 184
Receiver request fields 184
receiver response field 85, 107, 118, 132, 189
Receiver response fields 85, 107, 119, 191
ReceiverIdentifier request fields 142
ReceiverIdentifier response fields 133
receiverList request field 72, 183
ReceiverList request fields 75, 184
receiverOptions request field 139
ReceiverOptions request fields 141
receiverOptions response field 130
ReceiverOptions response fields 132
Refund API operation 179
refundedAmount response field 107
refundFeeAmount response field 189
refundGrossAmount response field 189
refundHasBecomeFull response field 189
refundInfo response field 189
RefundInfo response fields 189
refundInfoList response field 188

RefundInfoList response fields 189
 refundNetAmount response field 189
 RefundRequest fields 183
 RefundRequest message 181
 RefundResponse fields 188
 RefundResponse message 185
 refundStatus response field 190
 refundTransactionStatus response field 190
 req uestEnvelope request field 72
 requestEnvelope request field 115, 127, 139, 161, 173, 184, 200, 212, 221
 RequestEnvelope request fields 77, 100, 116, 127, 143, 155, 162, 173, 185, 203, 212, 222
 requestEnvelope request fields 154
 requestEnvelope response field 130
 requireShippingAddressSelection request field 141
 requireShippingAddressSelection response field 131
 responseEn velope response field 188
 ResponseEnvelope fields 87, 89, 109, 111, 120, 122, 133, 135, 143, 145, 156, 158, 168, 170, 174, 177, 192, 194, 207, 209, 216, 219, 224, 226
 responseEnvelope PPFault field 88, 110, 121, 134, 144, 157, 169, 176, 193, 208, 218, 225
 responseEnvelope response field 81, 104, 118, 166, 174, 205, 214, 223
 responseEnvelope response fields 156
 return_url IPN variable 235
 returnUrl request field 72
 returnUrl request fields 154
 returnUrl response field 104, 166
 reverse_all_parallel_payments_on_error IPN variable 234
 reverseAllParallelPaymentsOnError request field 72
 reverseAllParallelPaymentsOnError response field 104

S

sender request field 73
 sender_email IPN variable 234
 senderEmail request field 73
 senderEmail request fields 154
 senderEmail response field 104, 166
 senderFees response field 82, 214
 SenderIdentifier request fields 77
 senderOptions request field 139
 SenderOptions request fields 141
 senderOptions response field 130
 SenderOptions response fields 131

senderTransactionID response field 107
 senderTransactionStatus response field 107
 SetPaymentOptionsRequest fields 139
 SetPaymentOptionsResponse message 137, 143
 SetPaymentOptionsResponse operation 137
 setTrigger JavaScript function 231
 severity field 85, 89, 111, 119, 122, 135, 145, 158, 170, 176, 191, 194, 209, 218, 226
 shareAddress request field 141
 shareAddress response field 131
 sharePhoneNumber request field 141
 sharePhoneNumber response field 131
 shippingAddressId request field 139
 shippingAddressId response field 130
 startFlow JavaScript function 231
 startingDate request fields 154
 startingDate response field 166
 state response field 167, 223
 status IPN variable 233
 status response field 105, 166
 subdomain field 85, 89, 111, 119, 122, 135, 145, 158, 170, 176, 191, 194, 209, 218, 226

T

timestamp field 87, 89, 109, 111, 120, 122, 133, 135, 143, 145, 156, 158, 168, 170, 175, 177, 192, 194, 207, 209, 217, 219, 224, 226
 to response field 84, 216
 totalOfAllRefunds response field 190
 totalShipping request field 142
 totalShipping response field 132
 totalTax request field 142
 totalTax response field 132
 trackingId IPN variable 235
 trackingId request field 73, 184
 trackingId response field 105
 transaction_type IPN variable 233, 236
 transaction[n].amount IPN variable 235
 transaction[n].id IPN variable 234
 transaction[n].id_for_sender IPN variable 234
 transaction[n].invoiceId IPN variable 235
 transaction[n].is_primary_receiver IPN variable 235
 transaction[n].receiver IPN variable 235
 transaction[n].refund_account_charged IPN variable 235
 transaction[n].refund_amount IPN variable 235
 transaction[n].refund_id IPN variable 234

transaction[n].status IPN variable 234
transaction[n].status_for _sender_txn IPN variable 234
transactionId request field 184
transactionId response field 107
transactionStatus response field 107
type response field 84, 167, 216, 223

U

useCredentials request field 77