## Contents

Preface ................................................................. 5  
  Purpose .......................................................... 5  
  Audience ......................................................... 5  

Chapter 1  Overview to UATP Authorization ..................... 7  

Chapter 2  PaySOAP API Architecture ............................ 9  
  Services Architecture .......................................... 9  
  PayPal WSDL/XSD Schema Definitions. .......................... 10  
  API Concepts and Terminology ................................ 10  
  Security .......................................................... 11  
  SOAP RequesterCredentials: Username, Password, Signature, and Subject ........................................... 12  
  SOAP Service Endpoints. ....................................... 13  
  SOAP Implementation .......................................... 13  
  SOAP Message Style: doc-literal ................................ 13  
  SOAP Request Envelope ........................................ 13  
  Request Structure .............................................. 14  
  Response Structure ............................................ 15  
  Error Responses. ................................................ 17  
  CorrelationID for Reporting Problems to PayPal ............. 18  
  PayPal SOAP API Definitions ................................... 18  
  Character Encoding, Data Types and Formats, and Currencies ................................................................. 19  

Chapter 3  Core eBL Request/Response Structures ................. 21  
  Diagram of core eBL Types ..................................... 21  
  AbstractRequestType ........................................... 21  
    Fields .................................................................. 22  
  AbstractResponseType .......................................... 22  
    Fields .................................................................. 22  

Chapter 4  UATP API Descriptions .................................... 25  
  UATP Authorization Using the SOAP API ..................... 25
Preface

This specification details PayPal Universal Air Travel Plan (UATP) Authorization.

Purpose

The purpose of this specification is to define the Authorization interface PayPal is developing for use by the airline industry so that customers can choose PayPal as a payment option for purchasing airline tickets. This particular feature is specific to PayPal’s unique partner in the airline industry, Universal Air Travel Plan (UATP). It extends PayPal’s Name-Value pair or SOAP API interface to support the UATP number and expiration date as well as the approval code indicating the success of an authorization.

Audience

This specification is for use by UATP’s Authorization System.

Revision History

Revision history for Airline UATP Authorization Interface Specification.

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>May 2010</td>
<td>Added error code 10631.</td>
</tr>
<tr>
<td>December 2006</td>
<td>Created.</td>
</tr>
</tbody>
</table>
PayPal’s Airline product offers PayPal as a tender type to customers purchasing tickets from airline merchant websites. The PayPal Airline UATP Authorization specification describes the parameters that UATP’s Authorization System uses in the call to authorize PayPal transactions. It also describes the response fields and error codes.

Figure 1.1 summarizes airline UATP authorization.

**FIGURE 1.1 Authorization Summary**

Authorizations to be processed by PayPal include a UATPNumber in the format 1112nnnnnnnnnn. UATP’s Authorization System sends these authorizations to PayPal formatted as described in Chapter 4, “UATP API Descriptions.” You can use Name-Value pair or PayPal’s SOAP API for sending the authorization request. Upon success, PayPal returns an AuthorizationCode in the response. UATP’s Authorization System then sends the response to the airline merchant.
The PayPal SOAP API provides programmatic access to PayPal features and services. Developers can build custom applications, tools, and services that correspond to the same services and tools available through the main PayPal website, https://www.paypal.com/. Typical applications include searching for transactions, paying en masse, and making refunds. The API is based on open standards known collectively as “Web Services,” which include the Simple Object Access Protocol (SOAP), Web Services Definition Language (WSDL), and the XML Schema Definition language (XSD). These standards are supported by a wide range of development tools on a variety of platforms.

**Services Architecture**

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

*Figure 2.1 PayPal SOAP High-level Diagram*

In an object-oriented processing model, the interface to SOAP requests/responses is an object in your application’s native programming language. Your third-party SOAP client generates
business-object interfaces and network stubs from PayPal-provided WSDL and XSD files that specify the PayPal SOAP message structure, its contents, and the PayPal API service bindings. A business application works with data in the form of object properties to send and receive data by calling object methods. The SOAP client handles the details of building the SOAP request, sending it to the PayPal service, and converting the response back to an object.

**PayPal WSDL/XSD Schema Definitions**

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

<table>
<thead>
<tr>
<th>TABLE 2.1 Location of PayPal WSDL and XSD Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>Development and Test with the PayPal Sandbox API Service</td>
</tr>
<tr>
<td>eBL Base Components and Component Types</td>
</tr>
<tr>
<td><a href="https://www.sandbox.paypal.com/wsdl/CoreComponentTypes.xsd">https://www.sandbox.paypal.com/wsdl/CoreComponentTypes.xsd</a></td>
</tr>
<tr>
<td>Production with Live PayPal Web Services API Service</td>
</tr>
<tr>
<td>eBL Base Components and Component Types</td>
</tr>
<tr>
<td><a href="http://www.paypal.com/wsdl/CoreComponentTypes.xsd">http://www.paypal.com/wsdl/CoreComponentTypes.xsd</a></td>
</tr>
</tbody>
</table>

**API Concepts and Terminology**

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

<table>
<thead>
<tr>
<th>TABLE 2.2 Basic PayPal API Set-up Concepts and Terminology</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Term</strong></td>
</tr>
<tr>
<td>API Calls</td>
</tr>
<tr>
<td>API Certificate</td>
</tr>
</tbody>
</table>
PayPal SOAP API Architecture
Services Architecture

PayPal SOAP API Architecture

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

1. A required API username (Username field) and API password (Password field)
2. A third required authentication mechanism, which is either one of the following:
   - Client-side request signing via a PayPal-issues API Certificate
   - Request authentication via an API Signature included in the request (Signature field)
3. An optional third-party authorization to make the API call on some other account’s behalf (the optional Subject field).
4. Secure Sockets Layer (SSL) data transport
A failure of authenticated security at any one of these levels denies access to the PayPal SOAP API service.

**SOAP RequesterCredentials: Username, Password, Signature, and Subject**

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

In the SOAP request header, your SOAP client must set theUsername, Password elements to pass an API username/password combination. In addition, you can set the Signature or Subject elements to specify your API signature string and an optional third-party account email address for authentication. The following is a partial example of the RequesterCredentials elements required for all SOAP requests. For a correlation of these elements to the generic structure of an entire SOAP request, see “SOAP Request Envelope” on page 13.

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

where:

**Table 2.3 RequesterCredentials Authentication Elements in SOAP Header**

<table>
<thead>
<tr>
<th>Element</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Username&gt;</td>
<td>api_username</td>
<td>Your API username, which is auto-generated by PayPal when you apply for a digital certificate to use the PayPal SOAP API. You can see this value on <a href="https://www.paypal.com/">https://www.paypal.com/</a> in your Profile under API Access &gt; API Certificate Information.</td>
</tr>
<tr>
<td>&lt;Password&gt;</td>
<td>api_password</td>
<td>Your API password, which you specify when you apply for a digital certificate to use the PayPal SOAP API.</td>
</tr>
<tr>
<td>&lt;Signature&gt;</td>
<td>api_signature</td>
<td>Your API signature, if you use one instead of an API Certificate.</td>
</tr>
</tbody>
</table>
SOAP Service Endpoints

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

### TABLE 2.4 SOAP Service Endpoints

<table>
<thead>
<tr>
<th>Authentication Mechanism</th>
<th>Live Production Endpoint</th>
<th>Test (Sandbox) Endpoint</th>
</tr>
</thead>
<tbody>
<tr>
<td>API Signature</td>
<td><a href="https://api-3t.paypal.com/2.0/">https://api-3t.paypal.com/2.0/</a></td>
<td><a href="https://api-3t.sandbox.paypal.com/2.0/">https://api-3t.sandbox.paypal.com/2.0/</a></td>
</tr>
<tr>
<td>API Certificate</td>
<td><a href="https://api.paypal.com/2.0/">https://api.paypal.com/2.0/</a></td>
<td><a href="https://api.sandbox.paypal.com/2.0/">https://api.sandbox.paypal.com/2.0/</a></td>
</tr>
</tbody>
</table>

SOAP Implementation

This section contains information about the PayPal SOAP implementation.

SOAP Message Style: doc-literal

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

SOAP Request Envelope

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

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

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

General Structure of PayPal API SOAP Request

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

FIGURE 2.2 Diagram of SOAP Request Envelope

[Diagram of SOAP Request Envelope]
<specific_api_name_Request>
  <Version xmlns=urn:ebay:apis:eBLBaseComponents">service_version
  </Version>
  <required_orOptional_fields xsi:type="some_type_here">data
  </required_orOptional_fields>
</specific_api_name_Request>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

**TABLE 2.5 Annotation of Generic SOAP Request**

<table>
<thead>
<tr>
<th>Lines</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>12, 13</td>
<td>The <code>&lt;Username&gt;</code> and <code>&lt;Password&gt;</code> fields are part of the PayPal SOAP API <code>&lt;RequesterCredentials&gt;</code> security authentication mechanism you must construct for every SOAP request header. For details, see “SOAP RequesterCredentials: Username, Password, Signature, and Subject” on page 12.</td>
</tr>
<tr>
<td>14</td>
<td>The <code>&lt;Signature&gt;</code> element should include your API signature string if that is the kind of API credential you are using. For more details, see Table 2.3, “RequesterCredentials Authentication Elements in SOAP Header,” on page 12.</td>
</tr>
<tr>
<td>15</td>
<td>The <code>&lt;Subject&gt;</code> element can specify a third-party PayPal account by whom you are authorized to make this request. For more details, see Table 2.3, “RequesterCredentials Authentication Elements in SOAP Header,” on page 12.</td>
</tr>
<tr>
<td>19 through 27</td>
<td>The SOAP request for every PayPal API follows this element naming pattern. The API’s specific name is appended with Req, and in this element the specific_api_name_Request is nested. Each specific_api_name_Request has a corresponding specific_api_name_RequestType.</td>
</tr>
<tr>
<td>22</td>
<td>The number of the PayPal SOAP API version is required on each SOAP request. This version number is the value of ns:version in <a href="https://www.paypal.com/wsdl/PalPalSvc.wsdl">https://www.paypal.com/wsdl/PalPalSvc.wsdl</a></td>
</tr>
<tr>
<td>24</td>
<td>For details about required and optional elements and values for specific requests, see the description of individual APIs.</td>
</tr>
</tbody>
</table>

**Response Structure**

The following is an annotated description of the structure of a SOAP response from the PayPal API where response is Success:
<?xml version="1.0"?>
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:cc="urn:ebay:apis:CoreComponentTypes"
    xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
    xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/12/secext"
    xmlns:ebl="urn:ebay:apis:eBLBaseComponents"
    xmlns:ns="urn:ebay:api:PayPalAPI">
    <SOAP-ENV:Header>
        <Security
            xmlns="http://schemas.xmlsoap.org/ws/2002/12/secext"
            xsi:type="wsse:SecurityType" />
        <RequesterCredentials xmlns="urn:ebay:api:PayPalAPI"
            xsi:type="ebl:CustomSecurityHeaderType">
            <Credentials
                xmlns="urn:ebay:apis:eBLBaseComponents"
                xsi:type="ebl:UserIdPasswordType" />
        </RequesterCredentials>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body id="_0">
        <specific_api_name_Response
            xmlns="urn:ebay:api:PayPalAPI">
            <Timestamp
                xmlns="urn:ebay:api:PayPalAPI" date="Time in UTC/GMT" />
            <Ack
                xmlns="urn:ebay:apis:eBLBaseComponents">Success</Ack>
            <Version
                xmlns="urn:ebay:apis:eBLBaseComponents">
                serviceVersion
            </Version>
            <CorrelationId
                xmlns="urn:ebay:apis:eBLBaseComponents">
                applicationCorrelation
            </CorrelationId>
            <Build
                xmlns="urn:ebay:apis:eBLBaseComponents">
                api_build_number
            </Build>
            <elements_for_specific_api_response>
                data
            </elements_for_specific_api_response>
        </specific_api_name_Response>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
Error Responses

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

- ShortMessage
- LongMessage
- ErrorCode

For a description of the contents of these elements, see “Error Codes and Messages” on page 31.”

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

Example of SOAP Error Response: Bad Username or Password

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

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

PayPal SOAP API Definitions

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

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

- Enumerations: Enumerations are defined directly in the PayPal API schema.
Troubleshooting information: The PayPal API returns information about elements that trigger errors.

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

**NOTE:** eBL defines many structures that are specific to processing auctions. PayPal’s SOAP schema includes these definitions to maintain compatibility with eBay’s SOAP and for possible future joint use of SOAP across both eBay and PayPal. The material in this book focuses only on those SOAP definitions pertinent to use of the PayPal SOAP API.

### Character Encoding, Data Types and Formats, and Currencies

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

#### UTF-8 Character Encoding

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

#### Date/Time Formats

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

#### Core Currency Amount Data Type

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

1. The `currencyID` attribute is required.
2. The amount must have two decimal places.
3. The decimal separator must be a period (".").
4. You must not use any thousands separator.
5. `BasicAmountType` has a data type of `ebl:CurrencyCodeType`, which defines a large number of different currency codes. However, for your processing to succeed, you must set `currencyCode` to one of the values from the first column of "Core Currency Amount Data Type" on page 19. Some APIs support only a subset of these currencies.

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

```
<Amount currencyID="currencyCode">3.00</Amount>
```
Core eBL Request/Response Structures

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

- AbstractRequestType
- AbstractResponseType

Diagram of core eBL Types

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

**Figure 3.1 Core eBL Types**

**AbstractRequestType**

AbstractRequestType is the base type definition of all PayPal SOAP requests.
**IMPORTANT:** You must set the value of the **Version** element for all requests to any PayPal API.

**Fields**

**Table 3.1 AbstractRequestType Fields**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Data Type</th>
<th>Allowable Values</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>A string representing the version of the PayPal Schema, specifically, the value the version attribute of the <code>wsdl:definitions</code> element. For the location of the PayPal Schema, see “PayPal WSDL/XSD Schema Definitions” on page 10.</td>
<td><code>xs:string</code></td>
<td>See the the PayPal Schema.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**AbstractResponseType**

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

**Fields**

For error codes and messages and their meanings, see “Error Codes and Messages” on page 31.”

**Table 3.2 AbstractResponseType Fields**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Data Type</th>
<th>Possible Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timestamp</td>
<td>The date and time (UTC/GMT) the response was generated by the PayPal SOAP service as a result of processing a request.</td>
<td><code>xs:dateTime</code></td>
<td>Transaction-specific</td>
</tr>
</tbody>
</table>
### Table 3.2 AbstractResponseType Fields

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Data Type</th>
<th>Possible Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ack</td>
<td>A token representing the application-level acknowledgement code. Contains one of the following enumerated values:</td>
<td>xs:token</td>
<td>Success&lt;br&gt;Failure&lt;br&gt;SuccessWith&lt;br&gt;FailureWith&lt;br&gt;SuccessWithWarning&lt;br&gt;FailureWithWarning&lt;br&gt;CustomCode</td>
</tr>
</tbody>
</table>
  - Success: Request processing succeeded  
  - Failure: Request processing failed  
  - SuccessWithWarning: Request processing completed successfully but with some warning information in Errors that could be useful for the requesting application to process or record.  
  - FailureWithWarning: Request processing failed with some error and warning information that the requesting application should process to determine causes of failure.  
  - CustomCode: Reserved for internal or future use.  
| Correlation ID | A string that identifies the exact API request that produced this response  
  Character length: 41 characters maximum. | xs:string     | Transaction-specific                       |
| Errors        | Error code and messages for debugging a response.  
  - ErrorCode: token defining the error code number  
  - ShortMessage: string containing the error message  
  - LongMessage: string describing the error message  
  - ErrorParameters: error parameters from the credit card processor. Returned only with DoDirectPaymentResponse.  
  - SeverityCode: returns Error or Warning. | ns:ErrorType  | Transaction-specific                       |
| Version       | A string representing the version of the PayPal Schema, specifically, the value of the version attribute of the wsdl:definitions element.  
  For the location of the PayPal Schema, see “PayPal WSDL/XSD Schema Definitions” on page 10. | xs:string     | See the the PayPal Schema.                |
| Build         | A string representing the specific software build that processes the request and generates the response | xs:string     |                                           |
The airline merchant obtains the UATP number and expiration date from PayPal and passes these to UATP’s Authorization System. UATP’s Authorization System calls DoUATPAuthorizationRequest with the UATPNumber, UATPExpMonth, and UATPExpYear.

UATP’s Authorization System receives the DoUATPAuthorizationResponse and returns it to the airline merchant. A successful authorization includes the AuthorizationCode.

### UATP Authorization Using the SOAP API

The following table represents the composition of the SOAP API DoUATPAuthorizationRequestType and DoUATPAuthorizationResponseType elements. Elements required in the request are marked with an asterisk.

<table>
<thead>
<tr>
<th>DoUATPAuthorizationRequestType</th>
<th>DoUATPAuthorizationRequestType</th>
</tr>
</thead>
<tbody>
<tr>
<td>*Amount</td>
<td>TransactionID</td>
</tr>
<tr>
<td>*UATPNumber</td>
<td>Amount</td>
</tr>
<tr>
<td>*UATPExpMonth</td>
<td>AuthorizationCode</td>
</tr>
<tr>
<td>*UATPExpYear</td>
<td>InvoiceID</td>
</tr>
<tr>
<td>InvoiceID</td>
<td>UATPNumber</td>
</tr>
<tr>
<td></td>
<td>UATPExpMonth</td>
</tr>
<tr>
<td></td>
<td>UATPExpYear</td>
</tr>
</tbody>
</table>
Table 4.2 describes the **DoUATPAuthorizationRequest** parameters for the PayPal SOAP API.

**Table 4.2 DoUATPAuthorizationRequest Parameters for PayPal SOAP API**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Data Type</th>
<th>Allowable Values</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amount</td>
<td>Amount to authorize. Limitations: Value is a positive number which cannot exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,).</td>
<td>ebl:BasicAmountType</td>
<td>See description.</td>
<td>Yes</td>
</tr>
<tr>
<td>UATPNumber</td>
<td>Character length and limits: An alphanumeric string of 15 digits. The first 4 digits are the PayPal reserved bin 1112;</td>
<td>xs:string</td>
<td>See description.</td>
<td>Yes</td>
</tr>
<tr>
<td>UATPExpMonth</td>
<td>The month in which the UATP number expires. Character length and limits: Integer between 1 and 12 inclusive.</td>
<td>xs:int</td>
<td>See description.</td>
<td>Yes</td>
</tr>
<tr>
<td>UATPExpYear</td>
<td>The year in which the UATP number expires. Character length and limits: 4 numerical digits (2XXX).</td>
<td>xs:int</td>
<td>See description.</td>
<td>Yes</td>
</tr>
<tr>
<td>InvoiceID</td>
<td>An identifier used to query the TransactionSearch API about successful authorizations. It is replaced by the Ticket Document Number at the time of the Batch capture. Character length and limits: 127 single-byte alphanumeric characters.</td>
<td>xs:string</td>
<td>See description.</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 4.3 describes the **DoUATPAuthorizationResponse** fields for the PayPal SOAP API.

**Table 4.3 DoUATPAuthorizationResponse fields for PayPal SOAP API**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Data Type</th>
<th>Allowable Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amount</td>
<td>Amount specified in the request.</td>
<td>ebl:BasicAmountType</td>
<td>See description.</td>
</tr>
</tbody>
</table>
The following table represents the composition of the SOAP API `DoUATPAuthorizationRequestType` and `DoUATPAuthorizationResponseType` elements. Elements required in the request are marked with an asterisk.

**Table 4.4 Name-Value Pair DoUATPAuthorizationTypes**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Data Type</th>
<th>Allowable Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthorizationCode</td>
<td>Value that represents a unique authorization. This code should be used along with the UATPNumber, UATPExpMonth, and UATPExpYear to capture against this authorization. Character length and limits: 4 numeric digits.</td>
<td>xs:string</td>
<td>See description.</td>
</tr>
<tr>
<td>InvoiceID</td>
<td>The identifier sent in the call to <code>DoUATPAuthorizationRequest</code>. See InvoiceID description in Table 4.2.</td>
<td>xs:string</td>
<td>See description.</td>
</tr>
<tr>
<td>UATPNumber</td>
<td>Character length and limits: An alphanumeric string of 15 digits. The first 4 digits are the PayPal reserved bin 1112.</td>
<td>xs:string</td>
<td>See description.</td>
</tr>
<tr>
<td>UATPExpMonth</td>
<td>The month in which the UATP number expires. Character length and limits: Integer between 1 and 12 inclusive.</td>
<td>xs:int</td>
<td>See description.</td>
</tr>
<tr>
<td>UATPExpYear</td>
<td>The year in which the UATP number expires. Character length and limits: 4 numerical digits (2XXX).</td>
<td>xs:int</td>
<td>See description.</td>
</tr>
</tbody>
</table>
Table 4.5 describes the DoUATPAuthorizationRequest fields for the PayPal Name-Value Pair API.

### TABLE 4.5  DoUATPAuthorizationRequest Parameters for the PayPal Name-Value Pair API

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMT</td>
<td>Amount to authorize. Limitations: Value is a positive number which cannot exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,).</td>
<td>Yes</td>
</tr>
<tr>
<td>UATPNUMBER</td>
<td>Character length and limits: An alphanumeric string of 15 digits. The first 4 digits are the PayPal reserved bin 1112.</td>
<td>Yes</td>
</tr>
<tr>
<td>UATPEXPMONTH</td>
<td>The month in which the UATP number expires. Character length and limits: Integer between 1 and 12 inclusive.</td>
<td>Yes</td>
</tr>
<tr>
<td>UATPEXPYEAR</td>
<td>The year in which the UATP number expires. Character length and limits: 4 numerical digits (2XXX).</td>
<td>Yes</td>
</tr>
<tr>
<td>INVOICEID</td>
<td>An identifier used to query the TransactionSearch API about successful authorizations. It is replaced by the Ticket Document Number at the time of the Batch capture. Character length and limits: 127 single-byte alphanumeric characters.</td>
<td>Yes</td>
</tr>
<tr>
<td>CURRENCYCODE</td>
<td>A three-character currency code. Currently only USD is supported. Default: USD.</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 4.6 describes the DoUATPAuthorizationResponse fields for the PayPal Name-Value Pair API.

### TABLE 4.6  DoUATPAuthorizationResponse Fields for the PayPal Name-Value Pair API

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMT</td>
<td>Amount specified in the request.</td>
</tr>
<tr>
<td>AUTHORIZATIONCODE</td>
<td>Value that represents a unique authorization. This code should be used along with the UATPNUMBER, UATPEXPMONTH, and UATPEXPYEAR to capture against this authorization. Character length and limits: 4 numeric digits.</td>
</tr>
</tbody>
</table>
### Table 4.6 DoUATPAuthorizationResponse Fields for the PayPal Name-Value Pair API

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVOICEID</td>
<td>The identifier sent in the call to DoUATPAuthorizationRequest. See the InvoiceID description in Table 4.5.</td>
</tr>
<tr>
<td>UATPNUMBER</td>
<td>Character length and limits: An alphanumeric string of 15 digits. The first 4 digits are the PayPal reserved bin 1112.</td>
</tr>
<tr>
<td>UATPEXPMONTH</td>
<td>The month in which the UATP number expires. Character length and limits: Integer between 1 and 12 inclusive.</td>
</tr>
<tr>
<td>UATPEXPYEAR</td>
<td>The year in which the UATP number expires. Character length and limits: 4 numerical digits (2XXX).</td>
</tr>
</tbody>
</table>
5 Error Codes and Messages

The errors described in this chapter are returned by the PayPal SOAP and Name-Value Pair APIs. The chapter consists of the following sections:

- “Name-Value Pair Error Response Format” on page 31
- “General API Errors” on page 31
- “DoUATPAuthorization Errors” on page 33

Name-Value Pair Error Response Format

Name-value pair API error responses has the following general format. If the ACK value is Error or Warning, specific API response fields are not returned.

**TABLE 3.1 Format of an Error Response**

<table>
<thead>
<tr>
<th>Response Fields on Error</th>
<th>ACK=Error&amp;TIMESTAMP=date/timeOfResponse &amp;CORRELATIONID=debuggingToken&amp;VERSION=2.300000 &amp;BUILD=buildNumber&amp;L_ERRORCODE0=errorCode&amp; L_SHORTMESSAGE0=shortMessage &amp;L_LONGMESSAGE0=longMessage&amp;L_SEVERITYCODE0=severityCode</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Multiple errors can be returned. Each set of errors has a different numeric suffix, starting with 0 and incremented by one for each error.</td>
</tr>
</tbody>
</table>

General API Errors

**TABLE 3.2 General API Errors**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Short Message</th>
<th>Long Message</th>
<th>Correcting This Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>10002</td>
<td>Authentication/Authorization Failed</td>
<td>Username/Password is incorrect</td>
<td>This error can be caused by an incorrect API username, an incorrect API password, or an invalid API signature. Make sure that all three of these values are correct. For your security, PayPal does not report exactly which of these three values might be in error.</td>
</tr>
</tbody>
</table>
### General API Errors

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Short Message</th>
<th>Long Message</th>
<th>Correcting This Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>10002</td>
<td>Authentication/Authorization Failed</td>
<td>You do not have permissions to make this API call</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Authentication/Authorization Failed</td>
<td>Account is locked or inactive</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Internal Error</td>
<td>Internal Error</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Authentication/Authorization Failed</td>
<td>Internal Error</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Authentication/Authorization Failed</td>
<td>Account is not verified</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Authentication/Authorization Failed</td>
<td>This call is not defined in the database!</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Authentication/Authorization Failed</td>
<td>Token is not valid</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Restricted account</td>
<td>Account is restricted</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Authentication/Authorization Failed</td>
<td>Token is not valid</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Authentication/Authorization Failed</td>
<td>API access is disabled for this account</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Authentication/Authorization Failed</td>
<td>Client certificate is disabled</td>
<td></td>
</tr>
<tr>
<td>10002</td>
<td>Restricted account</td>
<td>Account is restricted</td>
<td></td>
</tr>
</tbody>
</table>
### DoUATPAuthorization Errors

#### TABLE 3.3 DoUATPAuthorization Error Messages

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Short Message</th>
<th>Long Message</th>
<th>Returned By API Call...</th>
<th>Correcting This Error...</th>
</tr>
</thead>
<tbody>
<tr>
<td>10001</td>
<td>Internal Error</td>
<td>Internal Error</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10001</td>
<td>Internal Error</td>
<td>Transaction failed due to internal error</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10004</td>
<td>Internal Error</td>
<td>Invalid argument</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10007</td>
<td>Permission denied</td>
<td>You do not have permissions to make this API call</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10009</td>
<td>Transaction refused</td>
<td>Account is locked or inactive</td>
<td>Retry the request at a later time or close order.</td>
<td></td>
</tr>
<tr>
<td>10010</td>
<td>Transaction refused because of an invalid argument. See additional error messages for details.</td>
<td>Invalid argument</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10600</td>
<td>Authorization voided.</td>
<td>Authorization is voided.</td>
<td>DoUATPAuthorization</td>
<td>Close the order or authorization.</td>
</tr>
<tr>
<td>10601</td>
<td>Authorization expired.</td>
<td>Authorization has expired.</td>
<td>DoUATPAuthorization</td>
<td>Close the order or authorization.</td>
</tr>
<tr>
<td>10602</td>
<td>Authorization completed.</td>
<td>Authorization has already been completed.</td>
<td>DoUATPAuthorization</td>
<td>Close the order or authorization.</td>
</tr>
<tr>
<td>10603</td>
<td>The buyer is restricted.</td>
<td>The buyer account is restricted.</td>
<td>DoUATPAuthorization</td>
<td>Contact the buyer.</td>
</tr>
<tr>
<td>10604</td>
<td>Authorization must include both buyer and seller.</td>
<td>Authorization transaction cannot be unilateral. It must include both buyer and seller to make an auth.</td>
<td>DoUATPAuthorization</td>
<td>Review the order to ensure customer and seller are both PayPal members.</td>
</tr>
<tr>
<td>10605</td>
<td>Unsupported currency.</td>
<td>Currency is not supported.</td>
<td>DoUATPAuthorization</td>
<td>Retry the request with a PayPal-supported currency.</td>
</tr>
<tr>
<td>10606</td>
<td>Buyer cannot pay.</td>
<td>Transaction rejected, please contact the buyer.</td>
<td>DoUATPAuthorization</td>
<td>Contact the buyer.</td>
</tr>
</tbody>
</table>
### Table 3.3: DoUATPAuthorization Error Messages

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Short Message</th>
<th>Long Message</th>
<th>Returned By API Call...</th>
<th>Correcting This Error...</th>
</tr>
</thead>
<tbody>
<tr>
<td>10608</td>
<td>Funding source missing.</td>
<td>The funding source is missing.</td>
<td>DoUATPAuthorization</td>
<td>Contact the buyer.</td>
</tr>
<tr>
<td>10609</td>
<td>Invalid transactionID.</td>
<td>Transaction id is invalid.</td>
<td>DoUATPAuthorization</td>
<td>Check the validity of the authorization ID and reattempt the request.</td>
</tr>
<tr>
<td>10610</td>
<td>Amount limit exceeded.</td>
<td>Amount specified exceeds allowable limit.</td>
<td>DoUATPAuthorization</td>
<td>Reattempt the request with a lower amount.</td>
</tr>
<tr>
<td>10618</td>
<td>Transaction already voided or expired.</td>
<td>Transaction has already been voided or expired.</td>
<td>DoUATPAuthorization</td>
<td>Close the order or authorization.</td>
</tr>
<tr>
<td>10620</td>
<td>Order has already been voided, expired or completed.</td>
<td>Order has already been voided, expired or completed.</td>
<td>DoUATPAuthorization</td>
<td>Close this order.</td>
</tr>
<tr>
<td>10621</td>
<td>Order has expired.</td>
<td>Order has expired.</td>
<td>DoUATPAuthorization</td>
<td>Close this order.</td>
</tr>
<tr>
<td>10622</td>
<td>Order is voided.</td>
<td>Order is voided.</td>
<td>DoUATPAuthorization</td>
<td>Close this order.</td>
</tr>
<tr>
<td>10623</td>
<td>Maximum number of authorization allowed for the order is reached.</td>
<td>Maximum number of authorization allowed for the order is reached.</td>
<td>DoUATPAuthorization</td>
<td>Capture this order.</td>
</tr>
<tr>
<td>10624</td>
<td>Duplicate invoice</td>
<td>Payment has already been made for this InvoiceID.</td>
<td>DoUATPAuthorization</td>
<td>Review the invoice ID and reattempt the request.</td>
</tr>
<tr>
<td>10625</td>
<td>Transaction refused because of an invalid argument. See additional error messages for details.</td>
<td>The amount exceeds the maximum amount for a single transaction.</td>
<td>DoUATPAuthorization</td>
<td>Reattempt the request with a lower amount.</td>
</tr>
</tbody>
</table>
### DoUATPAuthorization Error Messages

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Short Message</th>
<th>Long Message</th>
<th>Returned By API Call...</th>
<th>Correcting This Error...</th>
</tr>
</thead>
<tbody>
<tr>
<td>10626</td>
<td>Risk</td>
<td>Transaction refused due to risk model</td>
<td>DoUATPAuthorization</td>
<td>Contact the buyer.</td>
</tr>
<tr>
<td>10627</td>
<td>Transaction refused because of an invalid argument. See additional error messages for details.</td>
<td>The invoice ID field is not supported for basic authorizations</td>
<td>DoUATPAuthorization</td>
<td>The Invoice ID field can only be used with DoCapture.</td>
</tr>
<tr>
<td>10628</td>
<td>This transaction cannot be processed at this time. Please try again later.</td>
<td>This transaction cannot be processed at this time. Please try again later.</td>
<td>DoUATPAuthorization</td>
<td>Retry the request at a later time.</td>
</tr>
<tr>
<td>10630</td>
<td>Item amount is invalid.</td>
<td>Item amount is invalid.</td>
<td>DoUATPAuthorization</td>
<td>Check the item amount to ensure that it is not zero or negative.</td>
</tr>
<tr>
<td>10631</td>
<td>Processor Warning</td>
<td>The authorization is being processed</td>
<td>DoUATPAuthorization</td>
<td></td>
</tr>
<tr>
<td>11402</td>
<td>Invalid UATP Card number.</td>
<td>This transaction cannot be processed. UATP Card number validation failed.</td>
<td>DoUATPAuthorization</td>
<td></td>
</tr>
<tr>
<td>11403</td>
<td>UATP Card Expired.</td>
<td>This transaction cannot be processed. UATP Card expired.</td>
<td>DoUATPAuthorization</td>
<td></td>
</tr>
<tr>
<td>11404</td>
<td>Invalid UATP Card.</td>
<td>This transaction cannot be processed. No UATP card exist in the system for given card number and expiration.</td>
<td>DoUATPAuthorization</td>
<td></td>
</tr>
<tr>
<td>11405</td>
<td>Invalid Authorization code.</td>
<td>This transaction cannot be processed. No Authorization exists in the system for given UATP card and authorization code.</td>
<td>DoUATPAuthorization</td>
<td></td>
</tr>
<tr>
<td>Error Code</td>
<td>Short Message</td>
<td>Long Message</td>
<td>Returned By API Call...</td>
<td>Correcting This Error...</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>11406</td>
<td>Ineligible Merchant Account.</td>
<td>The transaction cannot complete successfully due to a failure in UATP number generation.</td>
<td>DoUATPAuthorization</td>
<td></td>
</tr>
</tbody>
</table>
Glossary

Authorization Code

Four-digit authorization code.

Airline

Refers to a carrier selling travel-related items on its own website.

Channel Partner

Term used synonymously with Partner for gateways, processors, and shopping carts who are directly involved in the boarding of new PayPal accounts or directing transaction volume to PayPal. The account that will be submitting batch files to create, manage, and report on other accounts.

Itinerary data

Itinerary data which is received through the Batch Capture interface.

Partner

See Channel Partner.

UATP

United Air Travel Plan, the travel association that functions similarly to VISA.

UATP BIN

First four digits of the PayPal/UATP card number (1112) that determine how the card is going to be processed by UATP.

UATP card

A credit card that airlines issue to corporations for corporate travel.
## Index

### A
- AbstractRequestType 21
- AbstractResponseType 21, 22
- ACK 31
- Ack 16, 23
  - success response 17
- Amount 26
- AMT 28
- API Certificate 10
- API password 12
- API Signature 11
- API username 12
  authentication for all SOAP requests 12
- AUTHORIZATIONCODE 28
- AuthorizationCode 27

### B
- BasicAmountType 19
- Batch capture 26, 28
- BUILD 31
- Build 23

### C
- Certificate 11, 13
  character encoding and character set
    UTF-8 19
- CoreComponentTypes.xsd 10
- CORRELATIONID 31
- CorrelationID 17, 18, 23
- CorrelationId 16
  currency amount data type 19
- CURRENCYCODE 28
  currencyID required attribute 19
- CustomCode 23

### D
- dateTime data type 19
- doc-literal SOAP messaging 13

### E
- eBL 18
- ErrorCode 17
- Errors 17, 23

### F
- Failure 23
- FailureWithWarning 23
  First-Party Access 11

### I
- INVOICEID 28, 29
- InvoiceID 26, 27

### L
- LongMessage 17

### P
- Password 11, 12, 14, 15
- PayPalSvc.wsdl 10

### R
- RequesterCredentials 12, 14, 15
  rpc-encoding not supported 13

### S
- security
  API password 12
  API username 12
  third-party authorization 13
- ShortMessage 17
- Signature 11, 12, 14, 15
Index

endpoint sfor 13
SOAP request authentication 12
Subject 11, 12, 13, 14, 15
Success 17, 23
SuccessWithWarning 23

T
Third-Party Access 11
third-party authorization 12, 13
Timestamp 16, 22
TRANSACTIONID 28
TransactionID 26
TransactionSearch API 26, 28

U
UATPEXPMONTH 29
UATPExpMonth 26, 27, 28
UATPEXPYEAR 29
UATPExpYear 26, 27, 28
UATPNUMBER 28, 29
UATPNumber 26, 27
Unicode 19
Username 11, 12, 14, 15
UTC/GMT 17, 19, 22

V
VERSION 31
Version 15, 22, 23

W
Web Services 9
WSDL 9, 10
  location of PayPal WSDL configuration file 10

X
XSD 9
  location for PayPal Web Services 10