Agenda

1. What’s new with managed payments
2. Payouts, transactions & orders
3. New APIs
4. Changes to existing APIs
Managed payments fundamentally changes eBay

We now have the ability to control and enhance all payments-related functions on eBay itself! No more logging into PayPal, or calling PayPal APIs – it is all happening on eBay.

<table>
<thead>
<tr>
<th>Managed Payments</th>
<th>Before</th>
<th>After</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Onboarding</strong></td>
<td>Register on eBay</td>
<td>Register on eBay, providing KYC and Bank information</td>
</tr>
<tr>
<td></td>
<td>Register on PayPal, providing KYC and Bank information</td>
<td></td>
</tr>
<tr>
<td><strong>Payouts</strong></td>
<td>Per order, to PayPal Account</td>
<td>Per day/week/month, all orders</td>
</tr>
<tr>
<td></td>
<td>See status on PayPal web/API</td>
<td>See status on eBay Seller Hub Payment Tab/API</td>
</tr>
<tr>
<td><strong>Refunds</strong></td>
<td>Through PayPal web/API</td>
<td>Through eBay Seller Hub/API</td>
</tr>
<tr>
<td></td>
<td>With cases through eBay returns and cancellations</td>
<td></td>
</tr>
</tbody>
</table>
# Critical definitions for new and changed APIs

<table>
<thead>
<tr>
<th>Orders</th>
<th>Transactions</th>
<th>Payouts</th>
</tr>
</thead>
</table>
| One or more items paid to the same seller in a single checkout | Funds flowing in from orders to a seller's payments balance on eBay  
Funds flowing out from returns, cancellations, and refunds from a seller's payments balance on eBay | A positive payments balance transferred to a seller’s bank account at the end of the day for all transactions that have cleared eBay's account |
New APIs

- Account API
  - Payments Program

- Fulfillment API
  - Refunds, Unique Order ID

- Finances API
  - Payouts & Transactions

Changed APIs

- Trading API
  - CompleteSale
  - GetOrders
  - GetOrderTransactions
  - GetSellingManagerEmailLog
  - GetSellingManagerSaleRecord
  - ReviseCheckoutStatus
  - ReviseSellingManagerSaleRecord
  - SendInvoice
  - GetItemTransactions
  - GetOrders
  - GetOrderTransactions
  - GetSellingManagerSaleRecord

- LMS
  - OrderReport
  - OrderAck
  - SetShipmentTrackingInfo
  - FeeSettlementReport
  - OrderReport
  - SoldReport

- Post Order API
  - Submit Cancellation Request
  - Check Cancellation Eligibility
  - getGuestPurchaseOrder
  - getPurchaseOrder
  - Search Cancellations

- getMyeBayBuying
- getMyeBaySelling
Account API
Payments Program
Eligible sellers can now opt-in to eBay’s managed payments offering.

This call returns the seller’s status for the specified payment program within a specific marketplace.

**Input**

This call does not use a request payload.

```
GET https://api.sandbox.ebay.com/sell/account/v1/payment_program/EBAY_US/EBAY_PAYMENTS
```

**Output**

A successful call returns an HTTP status of "200 OK" and a payload that contains the current status of the seller with regards to the specified payment program.

```
1 {  
2   "marketplaceId": "EBAY_US",  
3   "paymentProgramType": "EBAY_PAYMENTS",  
4   "status": "OPTED_IN",  
5   "wasPreviouslyOptedIn": "false"  
6 }
```
You can now track your funds on eBay

Phase one of the Finances API enable sellers to get payout status and balances for financial and reconciliation operations.
Demo
Get Payout Summary by Payee ID

Endpoint: GET/sell/finances/v1_alpha/payout_summary

Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>payee_id</td>
<td>string</td>
<td>true</td>
<td>payee id</td>
<td>token</td>
</tr>
<tr>
<td>filter</td>
<td>array:FilterField</td>
<td>false</td>
<td>payoutStatus [SUCCEEDED, RETRYABLE_FAILED, TERMINAL_FAILED, REVERSED] payoutDate</td>
<td>query</td>
</tr>
</tbody>
</table>

Get Payout Summary by Payee ID

Endpoint: GET/sell/finances/v1_alpha/payout_summary

Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>payee_id</td>
<td>string</td>
<td>true</td>
<td>payee id</td>
<td>token</td>
</tr>
<tr>
<td>filter</td>
<td>array:FilterField</td>
<td>false</td>
<td>payoutStatus [SUCCEEDED, RETRYABLE_FAILED, TERMINAL_FAILED, REVERSED] payoutDate</td>
<td>query</td>
</tr>
</tbody>
</table>

Get Payout Summary by Payee ID

Endpoint: GET/sell/finances/v1_alpha/payout_summary

Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>payee_id</td>
<td>string</td>
<td>true</td>
<td>payee id</td>
<td>token</td>
</tr>
<tr>
<td>filter</td>
<td>array:FilterField</td>
<td>false</td>
<td>payoutStatus [SUCCEEDED, RETRYABLE_FAILED, TERMINAL_FAILED, REVERSED] payoutDate</td>
<td>query</td>
</tr>
</tbody>
</table>
Get Payout Summary by Payee ID

Sample response

```json
{
    "amount": {
        "value": "(string)",
        "currency": "(enum) : [AED, AFN, ALL, AMD, AOA, ...]"
    },
    "transactionCount": "(integer)",
    "payoutCount": "(integer)"
}
```
# Get Payout by Payee ID

**Endpoint:** GET /sell/finances/v1_alpha/payout

## Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>payee_id</td>
<td>string</td>
<td>true</td>
<td>payee id</td>
<td>token</td>
</tr>
<tr>
<td>limit</td>
<td>string</td>
<td>false</td>
<td>pagination for lists of payouts, default value 20</td>
<td>query</td>
</tr>
<tr>
<td>v</td>
<td>string</td>
<td>false</td>
<td>pagination for lists of payouts, default value 0</td>
<td>query</td>
</tr>
<tr>
<td>sort</td>
<td>array:SortField</td>
<td>false</td>
<td>payoutDate</td>
<td>query</td>
</tr>
<tr>
<td>filter</td>
<td>array:FilterField</td>
<td>false</td>
<td>payoutStatus [SUCCEEDED, RETRYABLE_FAILED, TERMINAL_FAILED, REVERSED], payoutDate</td>
<td>query</td>
</tr>
</tbody>
</table>
Get Payout by Payee ID

Sample response

```json
{
    "total": "(integer)",
    "limit": "(integer)",
    "next": "(string)",
    "payouts": [
        {
            "amount": {
                "value": "(string)",
                "currency": "(enum) : [AED, AFN, ALL, AMD, AOA, ...]"
            },
            "payoutDate": "(string)",
            "transactionCount": "(integer)",
            "payoutStatus": "(enum) : [SUCCEEDED, REVERSED, RETRYABLE_FAILED, TERMINAL_FAILED]",
            "payoutId": "(string)",
            "payoutInstrument": {
                "accountLastFourDigits": "(string)",
                "instrumentType": "(string)",
                "institution": "(string)"
            }
        },
        {
            "offset": "(integer)",
            "href": "(string)",
            "prev": "(string)"
        }
    ]
}
```
Get Payout

Endpoint: GET /sell/finances/v1_alpha/payout/{payout_id}

Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>payout_id</td>
<td>string</td>
<td>true</td>
<td>payout id</td>
<td>path</td>
</tr>
</tbody>
</table>
Get Payout

Sample response

```json
{
   "amount": {
      "value": "(string)",
      "currency": "(enum) : [AED, AFN, ALL, AMD, AOA, ...]"
   },
   "payoutDate": "(string)",
   "transactionCount": "(integer)",
   "payoutStatus": "(enum) : [SUCCEEDED, REVERSED, RETRYABLE_FAILED, TERMINAL_FAILED]",
   "payoutId": "(string)",
   "payoutInstrument": {
      "accountLastFourDigits": "(string)",
      "instrumentType": "(string)",
      "institution": "(string)"
   }
}
```
## Get Transactions

**Endpoint:** GET /sell/finances/v1_alpha/transaction

### Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>payee_id</td>
<td>string</td>
<td>true</td>
<td>payee id</td>
<td>token</td>
</tr>
<tr>
<td>limit</td>
<td>string</td>
<td>false</td>
<td>pagination for lists of payouts, default value 20</td>
<td>query</td>
</tr>
<tr>
<td>offset</td>
<td>string</td>
<td>false</td>
<td>pagination for lists of payouts, default value 0</td>
<td>query</td>
</tr>
<tr>
<td>sort</td>
<td>array:SortField</td>
<td>false</td>
<td>transactionDate</td>
<td>query</td>
</tr>
<tr>
<td>filter</td>
<td>array:FilterField</td>
<td>false</td>
<td>transactionStatus [FUNDS_ON_HOLD, FUNDS_PROCESSING, FUNDS_AVAILABLE_FOR_PAYOUT, PAYOUT], transactionType [SALE, REFUND, CREDIT], transactionDate, buyerUsername, salesRecordReference, payoutId, transactionid</td>
<td>query</td>
</tr>
</tbody>
</table>
Sample response

```json
{
  "total": "(integer)",
  "limit": "(integer)",
  "transactions": [
    {
      "amount": {
        "value": "(string)",
        "currency": "(enum) : [AED, AFN, ALL, AMD, AOA, ...]"
      },
      "transactionType": "(enum) : [SALE, REFUND, CREDIT]",
      "transactionId": "(string)",
      "salesRecordReference": "(string)",
      "buyer": {
        "username": "(string)"
      },
      "payoutId": "(string)",
      "transactionStatus": "(enum) : [FUNDS_ON_HOLD, FUNDS_PROCESSING, FUNDS_AVAILABLE_FOR_PAYOUT, PAYOUT]",
      "transactionDate": "(string)"
    },
    {
      "next": "(string)",
      "offset": "(integer)",
      "href": "(string)",
      "prev": "(string)"
    }
  ]
}
```
## Get Transaction Summary by Payee ID

**Endpoint:** GET /sell/finances/v1_alpha/transaction_summary

### Parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>payee_id</td>
<td>string</td>
<td>true</td>
<td>payee id</td>
<td>token</td>
</tr>
<tr>
<td>filter</td>
<td>array:FilterField</td>
<td>false</td>
<td>transactionStatus [FUNDS_ON_HOLD, FUNDS_PROCESSING, FUNDS_AVAILABLE_FOR_PAYOUT, PAYOUT] transactionType [SALE, REFUND, CREDIT] transactionDate buyerUsername salesRecordReference Payoutid transactionid</td>
<td>query</td>
</tr>
</tbody>
</table>
Get Transaction Summary by Payee ID

Sample response

```json
{
    "refundCount": "(integer)",
    "refundAmount": {
        "value": "(string)",
        "currency": "(enum) : [AED, AFN, ALL, AMD, AOA, ...]"
    },
    "creditCount": "(integer)",
    "creditAmount": {
        "value": "(string)",
        "currency": "(enum) : [AED, AFN, ALL, AMD, AOA, ...]"
    },
    "onHoldAmount": {
        "value": "(string)",
        "currency": "(enum) : [AED, AFN, ALL, AMD, AOA, ...]"
    },
    "onHoldCount": "(integer)"
}
```
Fulfillment API
Seller Initiated Refunds
Fulfillment API: Seller Initiated Refunds

You can now process full or partial refunds outside of claims and cancellation flows!
Fulfillment API: Seller Initiated Refunds

Endpoint: GET /sell/fulfillment/v1/order/{order_id}/issue_refund

Parameters: N/A
Fulfillment API: Seller Initiated Refunds

Sample request

```json
{
    "reasonForRefund": "(enum)",
    "comment": "(string)",
    "refundItems": [
        {
            "legacyReference": {
                "legacyItemId": "(string)",
                "legacyTransactionId": "(string)"
            },
            "lineItemId": "(string)",
            "refundAmount": {
                "value": "(string)",
                "currency": "(enum) : [AED, AFN, ALL, AMD, AOA, ...]"
            }
        }
    ],
    "orderLevelRefundAmount": {
        "value": "(string)",
        "currency": "(enum) : [AED, AFN, ALL, AMD, AOA, ...]"
    }
}
```

Sample response

```json
{
    "refundId": "(string)",
    "refundStatus": "(enum)"
}
```
Unique Order IDs
We have added a unique order ID to orders, visible to both buyers and sellers, requiring changes to order APIs.
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Call Names (request)</th>
<th>Call Names (response)</th>
<th>API Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrderID</td>
<td>CompleteSale, GetOrders, GetOrderTransactions, GetSellingManagerEmailLog, GetSellingManagerSaleRecord, ReviseCheckoutStatus, ReviseSellingManagerSaleRecord, SendInvoice</td>
<td>AddOrder, GetItemTransactions, GetMyeBayBuying, GetMyeBaySelling, GetOrders, GetOrderTransactions, GetSellerTransactions</td>
<td>Trading API</td>
</tr>
<tr>
<td>ExtendedOrderID</td>
<td>GetItemTransactions, GetOrders, GetOrderTransactions, GetSellerTransactions</td>
<td>N/A</td>
<td>Trading API</td>
</tr>
<tr>
<td>orderId</td>
<td>getOrder, createShippingFulfillment, getShippingFulfillment, getShippingFulfillment</td>
<td>getOrder, getOrders</td>
<td>Fulfillment API</td>
</tr>
<tr>
<td>orderIds</td>
<td>getOrders</td>
<td>N/A</td>
<td>Fulfillment API</td>
</tr>
<tr>
<td>legacyOrderId</td>
<td>N/A</td>
<td>getOrder, getOrders</td>
<td>Fulfillment API</td>
</tr>
<tr>
<td>ExtendedOrderID</td>
<td>N/A</td>
<td>OrderReport</td>
<td>Large Merchant Services</td>
</tr>
<tr>
<td>OrderID</td>
<td>OrderAck, SetShipmentTrackingInfo, FeeSettlementReport, OrderReport, SoldReport</td>
<td></td>
<td>Large Merchant Services</td>
</tr>
<tr>
<td>legacyOrderId</td>
<td>Submit Cancellation Request, Check Cancellation Eligibility, Search Cancellations</td>
<td></td>
<td>Post Order API</td>
</tr>
<tr>
<td>purchaseOrderId</td>
<td>getGuestPurchaseOrder, getPurchaseOrder</td>
<td>getGuestPurchaseOrder, getPurchaseOrder</td>
<td>Buy Order API</td>
</tr>
<tr>
<td>legacyOrderId</td>
<td>N/A</td>
<td>getGuestPurchaseOrder, getPurchaseOrder</td>
<td>Buy Order API</td>
</tr>
</tbody>
</table>
Seize the Opportunity!

Give Feedback
Come upstairs to meet with the team
File a developer support ticket
Talk about it at the developer council

Start Now
We will be moving sellers to managed payments
Starting now will help you be ready for larger volumes over the next few years
You'll be able to offer new integrated payments capabilities in your application

Let us Help
If you don’t have a managed payments seller in your portfolio already, let us get you connected to one!
Thank you!