



Post Transaction Changes in 2014-15 – Playbook for CPs and Sellers

January 2015 / V1.1

Table of Contents

ABOUT THE AFTER-SALE APIS.....	4
OVERVIEW	4
TIMELINES FOR RELEASE ON EBAY SITES.....	5
<i>Release Schedule for the New After-Sale Experience (eBay Website UI)</i>	<i>5</i>
<i>Release Schedule for the Private Release After-sale APIs</i>	<i>5</i>
HANDLING AFTER-SALE ACTIVITIES – EXISTING AND NEW APIS	5
BUYER-INITIATED CANCELLATIONS	6
HANDLING CANCELLATIONS.....	6
HANDLING RETURNS.....	7
HANDLING RETURNS FOR SELLERS OPTED INTO HASSLE-FREE RETURNS.....	7
HANDLING RETURNS FOR SELLERS NOT OPTED INTO HASSLE-FREE RETURNS.....	9
PRODUCT FEATURES/CHANGES	11
ORDER LINE ACTIONS ROUTING	11
ORDER CANCELLATION	12
<i>Order Cancellation-Supported Integration Scenarios</i>	<i>14</i>
ORDER INQUIRY (INR – ITEM NOT RECEIVED)	17
<i>Order Inquiry-Supported Integration Scenarios</i>	<i>18</i>
RETURNS AND SNAD.....	19
<i>Order Returns-Supported Integration Scenarios</i>	<i>21</i>
ESCALATIONS.....	23
<i>Escalations Supported Integration Scenario(s)</i>	<i>23</i>
NEW AFTER-SALE API SET	25
MAKING A CALL.....	25
<i>Root Endpoints.....</i>	<i>25</i>
<i>Supported Formats</i>	<i>25</i>
<i>HTTP Headers.....</i>	<i>25</i>
ORDER CANCELLATION API	26
<i>POST /buy/order/v1/cancellation/cancel</i>	<i>26</i>
<i>GET /buy/order/v1/cancellation/checkEligibility.....</i>	<i>27</i>
<i>POST /buy/order/v1/cancellation/confirm</i>	<i>28</i>
<i>GET /buy/order/v1/cancellation/find</i>	<i>29</i>

Private Release – Do Not Distribute

Page 1 of 173

GET /buy/order/v1/cancellation/{cancelld}.....	30
POST /sell/order/v1/cancellation/approve.....	31
POST /sell/order/v1/cancellation/cancel.....	32
GET /sell/order/v1/cancellation/checkEligibility	33
GET /sell/order/v1/cancellation/find.....	33
POST /sell/order/v1/cancellation/reject.....	35
GET /sell/order/v1/cancellation/{cancelld}	35
ORDER INQUIRY API.....	37
POST /buy/order/v1/inquiry	37
POST /buy/order/v1/inquiry/closure	37
POST /buy/order/v1/inquiry/eligibility	38
POST /buy/order/v1/inquiry/escalation	44
POST /buy/order/v1/inquiry/message.....	45
POST /buy/order/v1/inquiry/refundConfirmation	45
GET /buy/order/v1/inquiry/{inquiryld}	46
GET /buy/order/v1/inquiry/{inquiryld}/history.....	49
POST /sell/order/v1/inquiry/escalation	51
POST /sell/order/v1/inquiry/message	52
POST /sell/order/v1/inquiry/refund.....	52
POST /sell/order/v1/inquiry/refundInfo.....	53
POST /sell/order/v1/inquiry/shipmentInfo	53
GET /sell/order/v1/inquiry/{inquiryld}.....	54
GET /sell/order/v1/inquiry/{inquiryld}/history	58
ORDER RESOLUTION API.....	59
POST /buy/order/v2/resolutioncasemanagement/appeal	59
POST /buy/order/v2/resolutioncasemanagement/closure.....	60
POST /buy/order/v2/resolutioncasemanagement/dispute	61
POST /buy/order/v2/resolutioncasemanagement/escalation.....	62
POST /buy/order/v2/resolutioncasemanagement/shipmentInfo	62
GET /buy/order/v2/resolutioncasemanagement/{caseld}	63
POST /sell/order/v2/resolutioncasemanagement/appeal.....	66
POST /sell/order/v2/resolutioncasemanagement/dispute.....	67
POST /sell/order/v2/resolutioncasemanagement/refund	68
POST /sell/order/v2/resolutioncasemanagement/returnaddress	69
GET /sell/order/v2/resolutioncasemanagement/{caseld}.....	69
ORDER RETURN API	74
GET /buy/order/v1/return/checkBillingAgreement	74
POST /buy/order/v1/return/create.....	74
POST /buy/order/v1/return/draft.....	81
GET /buy/order/v1/return/draft/{draftld}.....	84
PUT /buy/order/v1/return/draft/{draftld}.....	86
GET /buy/order/v1/return/eligibility	90
GET /buy/order/v1/return/estimate	105
GET /buy/order/v1/return/metadata/{siteMetadataKey}.....	110
GET /buy/order/v1/return/{returnld}	111

<i>POST /buy/order/v1/return/{returnId}/cancel</i>	116
<i>POST /buy/order/v1/return/{returnId}/escalate</i>	117
<i>POST /buy/order/v1/return/{returnId}/initiateShippingLabel</i>	119
<i>POST /buy/order/v1/return/{returnId}/markAsReceived</i>	120
<i>POST /buy/order/v1/return/{returnId}/markAsShipped</i>	121
<i>POST /buy/order/v1/return/{returnId}/printShippingLabel</i>	125
<i>GET /buy/order/v1/return/{returnId}/trackingHistory</i>	126
<i>GET /sell/order/v1/return/eligibility</i>	129
<i>GET /sell/order/v1/return/summary</i>	144
<i>GET /sell/order/v1/return/{returnId}</i>	149
<i>POST /sell/order/v1/return/{returnId}/authorize</i>	153
<i>POST /sell/order/v1/return/{returnId}/escalate</i>	156
<i>POST /sell/order/v1/return/{returnId}/issueRefund</i>	157
<i>POST /sell/order/v1/return/{returnId}/markAsReceived</i>	160
<i>POST /sell/order/v1/return/{returnId}/markAsShipped</i>	162
<i>GET /sell/order/v1/return/{returnId}/trackingHistory</i>	165
<i>POST /sell/order/v1/return/{returnId}/markRefundSent</i>	167
<i>POST /buy/order/v1/return/{returnId}/markRefundReceived</i>	170
<i>POST /buy/order/v1/return/{returnId}/sendMessage</i>	171
<i>POST /sell/order/v1/return/{returnId}/sendMessage</i>	172

About the After-Sale APIs

The new *after-sale* APIs described in this guide are provided for private release at this time to providers and sellers who have previously integrated with the Resolution Case Management API and Return Management API. These APIs will be fully operational in the production environment according to the timeline provided below. Most of these APIs will also be fully operational in the Sandbox. The public release of the new after-sale APIs will be rolled out in mid-Q4 of 2014. The function and flows described in the playbook will continue to apply for both the private and public releases. The after-sale APIs and flow are only available to US, UK, CA, DE, and AU sellers, and there are some differences in behavior between the US/UK sites and the CA/DE/AU sites. These differences are called out throughout this guide.

The Sandbox does not fully support initiating return and cancel requests via the web UI flow. To help you with end-to-end testing, you are welcome to use the new buy-side API calls, such as a buyer order cancel (POST /buy/order/v1/cancellation/cancel), to initiate buy-side requests in the Sandbox (and optionally, in Production). You can run these with RESTful API testing tools such as [Postman for Chrome](#) or [SoapUI](#), or via your own automated testing.

Overview

eBay is launching enhancements to the post-purchase experience, based on retail-standard concepts. The enhancements provide simpler self-service flows available across multiple devices. Post-purchase experiences will be referred to as “post-transaction” or “after-sale” throughout this document.

This document details several features that eBay will release in September 2014 and the steps that developers should take to integrate and adapt to after-sale API changes. The major new features include:

- **Order Line Actions Routing** - Simplifies the set of actions that customers can perform on an order line (transaction) level.
- **Order Cancellation** – Retail-standard cancel: Enables buyers *and* sellers to cancel paid and unpaid orders that haven’t shipped yet.
- **Order Inquiry (Item Not Received)** - Enhances the member self-service workflows to address the needs of members when their item has not arrived within the estimated delivery period. Note that this feature will not be available in the Sandbox initially.

Note: Buyers submit an Item Not Received (INR) request to inquire about where their order is and when it is going to be delivered; INR is an eBay specific term. The eBay after-sale APIs use the more standard industry term “order inquiry,” where the inquiry can be about delivery or other ordering lifecycle concerns.

- **Returns and SNAD** – When buyers need to get a refund or replacement for a received item, they will be able to initiate a return from any device.
- **Escalate** - Enables buyers and sellers to reach out to eBay Customer Support if a dispute arises in the after-sale self-service flow. Note that this feature will not be available in the Sandbox initially.

Important: The dates and details shared in this document are for your reference only, and will not be published to other parties who have not already integrated with existing after-sale APIs. If you are a Certified Provider or any other third party, and we have shared this document with you, we assume that you will use it for internal discussions and as a project implementation aid. Due to the fact that this is a highly complex and global initiative, project details, as well as their implementation timelines, can change on short notice.

Timelines for Release on eBay Sites

This section provides the schedule for the availability of the new experience and the related new APIs.

Release Schedule for the New After-Sale Experience (eBay Website UI)

- US, UK – September 2014
- AU, CA – November 2014
- DE – February 2015

Release Schedule for the Private Release After-sale APIs

- Order Cancellation API – September 23, 2014
- Order Return API – September 16, 2014
- Order Inquiry API – September 16, 2014
- Resolution Case Management API, v2 – September 16, 2014

Handling After-Sale Activities – Existing and New APIs

Existing APIs, specifically Trading, Resolution Case Management, and Return Management APIs, can still be used to handle after-sale activities such as order cancellations, returns, or cases. However, some new fields were introduced in the Trading API to handle new after-sale activities, and these new fields and the required new flow to handle after-sale activities are discussed in this playbook, including the minimum changes required to ensure a good buyer experience and minimize the risk of seller defects.

The new after-sale APIs were specifically designed to handle all after-sale activities, from start to finish. The Order Cancellation API handles all use cases and flow for buyer-initiated order cancellations. The Order Inquiry API and Order Resolution API handle all use cases and flow for Item Not Received (INR)

and Significantly Not As Described (SNAD) cases. The Order Return API handles all use cases and flow for seller returns.

Buyer-Initiated Cancellations

If you use an existing API (Trading, Resolution Case Management, or Return Management APIs) to handle after-sale activities such as cancellations, returns, or cases, you may need to make some changes due to the *new after-sale experience*. This section outlines the minimum changes needed for existing integrations to ensure a good buyer experience and minimize the risk of seller defects.

Handling Cancellations

With the rollout of the new after-sale experience, buyers can now request their own order cancellation, as long as it's within an hour of purchase and the seller hasn't shipped the item. While a buyer can initiate the cancellation, the seller still has to approve or reject it. See the [Improving the Experience, After the Sale](#) page to learn more about what's changing with the cancellation process.

To handle a buyer-initiated cancellation request, use the following flow:

1. To discover any pending cancellation requests for orders, do one of the following:
 - **(After-Sale API)** Use the `/sell/order/v1/cancellation/find` HTTP GET call in the Order Cancellation API to discover any pending cancellation request for an order. This call was designed to retrieve orders going through the cancellation flow. The `/sell/order/v1/cancellation/find` HTTP GET call has query parameter options in the call request to filter by order ID, user name, cancellation ID, and date range.
 - **(Trading API)** Use `GetOrders` (or `GetItemTransactions`, `GetOrderTransactions`, or `GetSellerTransactions`) in the Trading API. If a cancellation request was initiated by the buyer, the `Order.OrderStatus` value for the order will be returned as `'CancelPending'`. In addition to `'CancelPending'` being returned in the `Order.OrderStatus` field, the `Order.CancelStatus` field will also be returned and its initial value will be `'CancelRequested'`. The `Order.CancelStatus` value will change as the order goes through the cancellation flow. To specifically look for orders in the `'CancelPending'` status using `GetOrders`, the seller can pass the `'CancelPending'` value into the `OrderStatus` filter in the call request.
2. After the buyer initiates an order cancellation, the seller can do any of the following to accept or reject the request:
 - **(After-Sale API)** Use the `/sell/order/v1/cancellation/approve` or `/sell/order/v1/cancellation/reject` HTTP POST calls of the Order Cancellation API to approve or reject the order cancellation request. The cancellation ID is required in the input payload of either call to identify the order that is being cancelled. An ID is created for an order cancellation request once the buyer initiates the request.
 - **(Web flow)** Cancel the order through My eBay or in the Resolution Center.

- **(Trading API)** Create a seller-initiated mutual cancellation instead by using **AddDispute** in the Trading API. To do this in an **AddDispute** request, the seller passes the 'TransactionMutuallyCanceled' value into the **DisputeReason** field, as well as passing the appropriate and supported enumeration value into the **DisputeExplanation** field. Note that the buyer's order cancellation request may time out, but the end result will be that the order was cancelled by the seller at the request of the buyer.
- To reject a buyer's order cancellation request, the seller does nothing. The buyer's request will time out after three business days and the order cancellation will be automatically rejected.

Note: The new order cancellation Web flow and Order Cancellation API cancels an *entire order*. Due to this fact, sellers can only use the **AddDispute** call in the Trading API to cancel a single-line-item order, since this call only supports the cancellation of a specific order line item, and not multiple line items within a multiple-line-item order.

Handling Returns

How returns are handled depends on whether the seller is opted in to [hassle-free returns](#). With hassle-free returns, when a buyer sees and reports an item as "Significantly Not As Described" (SNAD), the report is automatically handled as a return request as soon as the buyer elects to "Return this item" in My eBay. The Return Management API is used to handle hassle-free returns programmatically. If the seller is not opted in to hassle-free returns, the seller can dispute the SNAD status of the order, in which case an eBay Buyer Protection SNAD case is actually created.

Handling Returns for Sellers Opted into Hassle-free Returns

Hassle-free returns are only available to US and UK sellers. When a seller is opted in to hassle-free returns, the buyer may seamlessly create a return request for an order line item that is "significantly not as described". If eligible, the seller opts into hassle-free returns through Return Preferences in My eBay. There is also an option in Trading API's **GetUserPreferences** call to see if a seller's account is opted in to hassle-free returns, but there is not a programmatic way to opt in to hassle-free returns. Once opted into hassle-free returns, the seller can set up automation rules in Return Preferences. These rules include the following:

- **Return Shipping Address rules** – by default, the buyer will return an item to the seller's default return address that is configured for the account; however, it is possible for the seller to customize where an item is returned based on purchase price, return reason, product category, and/or based on a custom item list.
- **Keep Item and Issue Refund rules** – by default, the buyer is expected to return the item to the seller; however, it is possible for the seller to create a rule that allows the buyer to keep the item while still getting a refund. Each rule is based on purchase price, return reason, product category, and/or based on a custom item list.

- **Keep Item and Send Replacement Item rules** – by default, the buyer is expected to return the item to the seller; however, it is possible for the seller to create a rule that allows the buyer to keep the original item and get a replacement item. Each rule is based on purchase price, return reason, product category, and/or based on a custom item list.

To handle a buyer-initiated, hassle-free return request, use the following flow:

1. To discover any pending return requests for order line items, do one of the following:
 - **(After-Sale API)** Use the **/sell/order/v1/return/summary** HTTP GET call in the Order Return API to discover any pending return request for an order line item. This call was designed to retrieve order line items going through the return flow. The **/sell/order/v1/return/summary** HTTP GET call has query parameter options in the call request to filter by order ID, line item ID, user name, return state, and date range.
 - **(Return Management API)** Use **getUserReturns** of the Return Management API to retrieve and process any pending return requests against the seller's account. The seller has filter options in the **getUserReturns** call request to find one or more return requests based on creation date, order ID, line item ID, or return status. The seller can also search for return request(s) initiated by a specific buyer. Optionally, the seller can use **getReturnDetail** of the Return Management API to retrieve detailed information about a specific return.
 - **(Trading API)** Use **GetOrders** (or **GetItemTransactions**, **GetOrderTransactions**, or **GetSellerTransactions**) in the Trading API. If a return request for an order line item was initiated by the buyer, a **ReturnStatus** value of 'ReturnRequestPendingApproval' will be returned under the **Transaction.Status** container. The **Transaction.Status.ReturnStatus** value will change as the order line item goes through the return flow.
2. Issue a full refund after receiving the returned item from the buyer using one of the following methods:
 - **(After-Sale API)** Use **/sell/order/v1/return/{returnid}/issueRefund** of the Order Return API to issue a refund to the buyer. When using **issueRefund**, a separate **itemizedRefundDetail** container is required for each fee type, including the item price, the original shipping cost, and a restocking fee (if one was set for the listing). A seller cannot charge the buyer a restocking fee for a SNAD item.
 - **(Return Management API)** Use **issueRefund** call of the Return Management API to issue a full refund to the buyer. When using **issueRefund**, a separate **itemizedRefund** container is required for each fee type, including the item price, the original shipping cost, and a restocking fee (if one was set for the listing). A seller cannot charge the buyer a restocking fee for a SNAD item.
 - **(Web flow)** Issue a refund to the buyer through My eBay.

Notes: The seller's account is reimbursed for the Final Value Fee as soon as a full refund is issued to the buyer. If the seller does not issue a full refund to the buyer within 6 business days after receiving

the returned item, eBay/PayPal will automatically issue a refund to the buyer and charge the seller's account.

Handling Returns for Sellers Not Opted into Hassle-free Returns

When a seller is not opted in to hassle-free returns, the buyer's request for return is not automatically approved. Since hassle-free returns are not yet available on the CA, DE, and AU sites, the sellers on these sites will always use the flow below.

To handle a buyer-initiated return request, use the following flow:

1. To discover any pending return requests for order line items, do one of the following:
 - **(After-Sale API)** Use the **/sell/order/v1/return/summary** HTTP GET call in the Order Return API to discover any pending return request for an order line item. This call was designed to retrieve order line items going through the return flow. The **/sell/order/v1/return/summary** HTTP GET call has query parameter options in the call request to filter by order ID, line item ID, user name, return state, and date range.
 - **(Return Management API)** Use **getUserReturns** of the Return Management API to retrieve and process any pending return requests against the seller's account. The seller has filter options in the **getUserReturns** call request to find one or more return requests based on creation date, order ID, line item ID, or return status. The seller can also search for return request(s) initiated by a specific buyer. Optionally, the seller can use **getReturnDetail** of the Return Management API to retrieve detailed information about a specific return.
 - **(Trading API)** Use **GetOrders** (or **GetItemTransactions**, **GetOrderTransactions**, or **GetSellerTransactions**) in the Trading API. If a return request for an order line item was initiated by the buyer, a **ReturnStatus** value of 'ReturnRequestPendingApproval' will be returned under the **Transaction.Status** container. The **Transaction.Status.ReturnStatus** value will change as the order line item goes through the return flow.
2. After the buyer initiates return request for an order line item, the seller can do any of the following as the next action:
 - Send a message to the buyer using one of the following methods:
 - **(After-Sale API)** Use **/sell/order/v1/return/{returnId}/sendMessage** HTTP POST call in the Order Return API to send a message to the buyer. The intent of this message may be to find out more information about why the item is being returned, or to discover exactly what the buyer is hoping for (a full refund, a partial refund, a replacement item).
 - **(Trading API)** Use **AddMemberMessageAAQToPartner** call to send a message to the buyer to find out more information about why the item is being returned, or to discover exactly what the buyer is hoping for (a full refund, a partial refund, a replacement item)

- (Web flow) Select the “Reply to Buyer” option through My eBay.

The next action (covered in this list) may be determined by the buyer’s response to the seller’s message

- Accept the return request from the buyer using one of the following methods:
 - (After-Sale API) Use `/sell/order/v1/return/{returnId}/authorize` HTTP POST call in the Order Return API to approve the return request. The seller should pass in a value of ‘APPROVE’ into the **decision** input payload field.
 - (Web flow) Select the “Accept the return” option through My eBay.

After accepting the return request, the seller would wait for the item to be returned by the buyer before issuing a refund to the buyer. If the seller is responsible for return shipping, the seller may provide the buyer with a shipping label.

- Send a full refund to the buyer using one of the following methods:
 - (After-Sale API) Use `/sell/order/v1/return/{returnId}/issueRefund` HTTP POST call in the Order Return API to issue a refund to the buyer. When using **issueRefund**, a separate **itemizedRefundDetail** container is required for each fee type, including the item price, the original shipping cost, and a restocking fee (if one was set for the listing). A seller cannot charge the buyer a restocking fee for a SNAD item.
 - (Web flow) Select the “Send a full refund” option through My eBay.

This action will close the return request. When this action is selected, the buyer is not expected to return the SNAD item to the seller.

- Send a partial refund to the buyer using one of the following methods:
 - (After-Sale API) Use `/sell/order/v1/return/{returnId}/issueRefund` HTTP POST call in the Order Return API to issue a partial refund to the buyer. When using **issueRefund** to issue a partial refund, one **itemizedRefundDetail** container should be used, and the value passed into the **itemizedRefundDetail.refundFeeType** field should be `ORDER_ADJUSTMENT`.
 - (Web flow) Select the “Offer a partial refund” option through My eBay.

Although not required, it is recommended that the seller contact the buyer to come to an agreement about a fair partial refund amount. Once the buyer receives the refund and is in agreement with the amount, that buyer can close the return request in My eBay or through the `/buy/order/v1/return/{returnId}/cancel` HTTP POST call of the Order Return API. If the buyer is not in agreement with the partial refund amount for the SNAD item, that buyer can escalate the return request in My eBay or through the `/buy/order/v1/return/{returnId}/escalate` HTTP POST call of the Order Return API. The result of this call is an eBay Buyer Protection SNAD case, which can then be managed by

the seller in the Resolution Case Management API (old) or in the Order Resolution API (new).

- Decline the buyer's return request using one of the following methods:
 - **(After-Sale API)** Use `/sell/order/v1/return/{returnId}/authorize` HTTP POST call in the Order Return API to decline the return request. The seller should pass in a value of 'DECLINE' into the **decision** input payload field to decline the return request.
 - **(Web flow)** Select the "Decline the return" option through My eBay.
 - Do nothing and the return request will time out and be automatically rejected. There is no seller defect unless the buyer escalates the return request to a eBay Buyer Protection SNAD case and eBay customer support rules in the buyer's favor.

If the seller declines the buyer's return request, the return request will be closed. If the buyer wishes to dispute this action, that buyer can escalate the SNAD item return request in My eBay or through the `/buy/order/v1/return/{returnId}/escalate` HTTP POST call of the Order Return API. The result of this call is an eBay Buyer Protection SNAD case, which can then be managed by the seller in the Resolution Case Management API (old) or in the Order Resolution API (new).

- Ask eBay Customer Support to step in and help using one of the following methods:
 - **(After-Sale API)** Use `/sell/order/v1/return/{returnId}/escalate` HTTP POST call of the Order Return API to ask eBay Customer Support to step in an help. It is suggested that the seller contact the buyer to try and work something out before using this action. This action is used when the seller feels that the buyer is not adhering to the return policy.
 - **(Web flow)** Select the "Ask us to step in and help" option through My eBay.

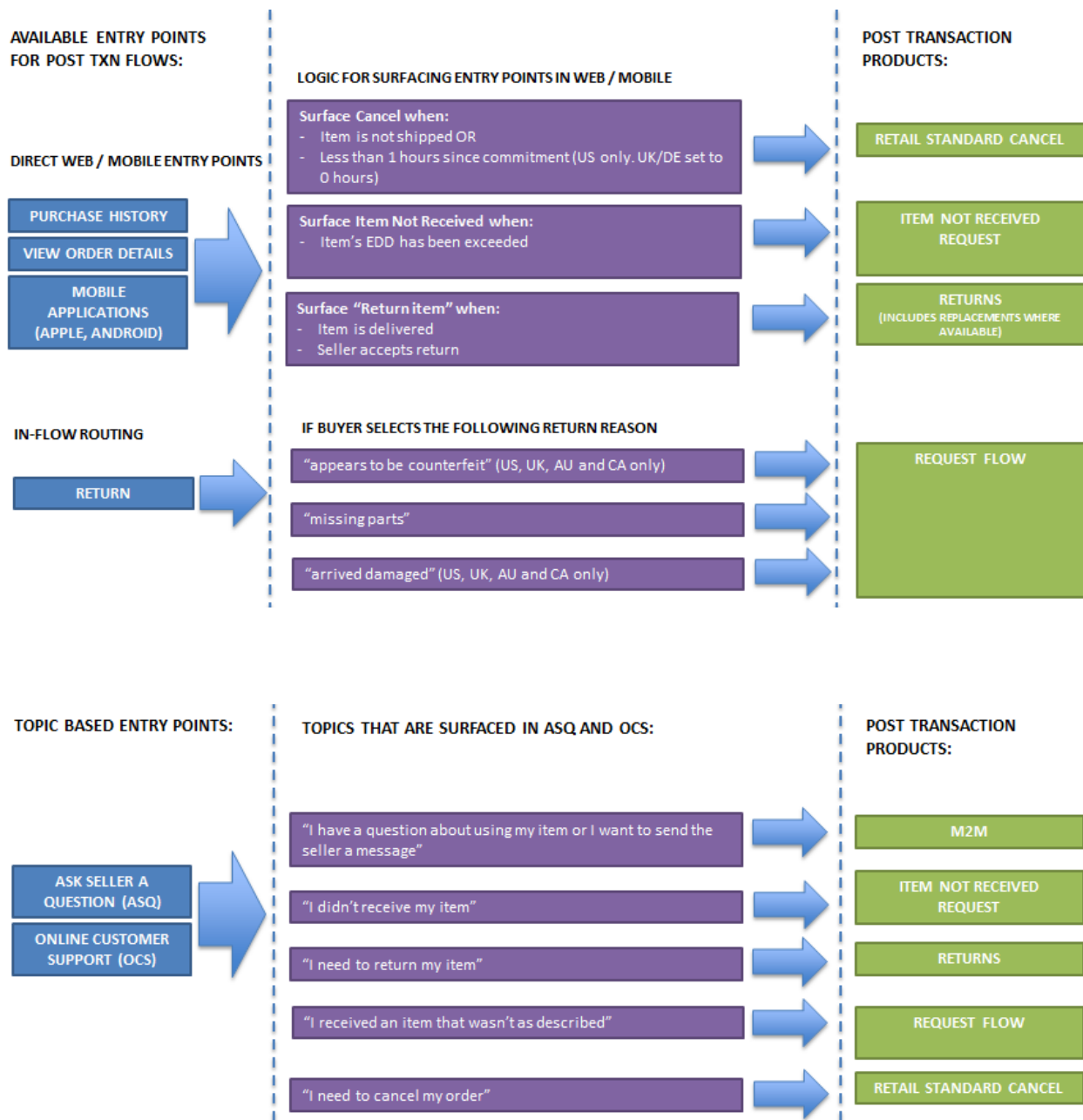
Product Features/Changes

This section gives an overview of how post-sale activities have changed, and includes descriptions of the new options available in the existing APIs (Trading, Return Management, and Resolution Case Management), and the new After-Sale APIs that are available (Order Cancellation, Order Inquiry, Order Resolution, Order Return). If you are looking for the shortest upgrade or migration path for your existing after-sale integration, you can use the prior section as a guideline and then read the corresponding sections below.

Order Line Actions Routing

API changes: NO IMPACT (There is no current or new public API providing order line level actions)

A buyer will have the following after-sale entry points available:



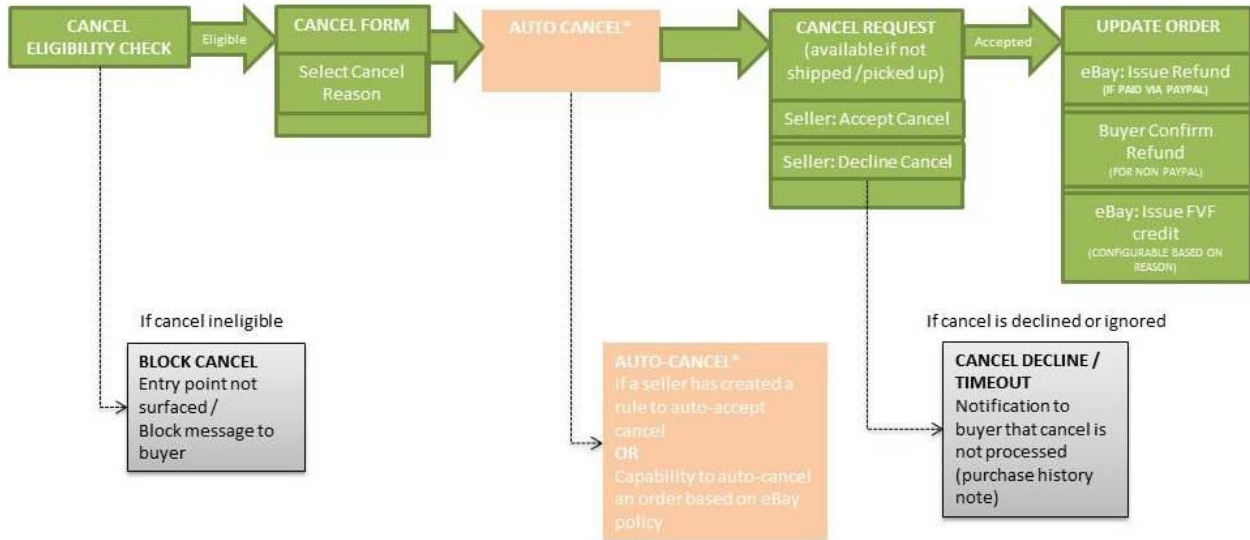
Order Cancellation

API changes summary: No change to the dispute-related calls in the Trading API; a new *Order Cancellation API* is introduced; the order management calls in the Trading API (**GetOrders**, **GetItemTransactions**, **GetOrderTransactions**, **GetSellerTransactions**) now returns a **CancelStatus** field set to "CancelRequested" for new cancellation requests.

A buyer will be able to initiate an order cancellation during a specified timeframe after checkout and before the order is shipped. The timeframe will be defined by eBay policy and can differ between sites.

Note: Buyer-initiated cancellations are only supported at the order level. Order line item level cancellations will be supported in a future release (planned for 2015).

Figure 1- Buyer Initiated Order Cancellation

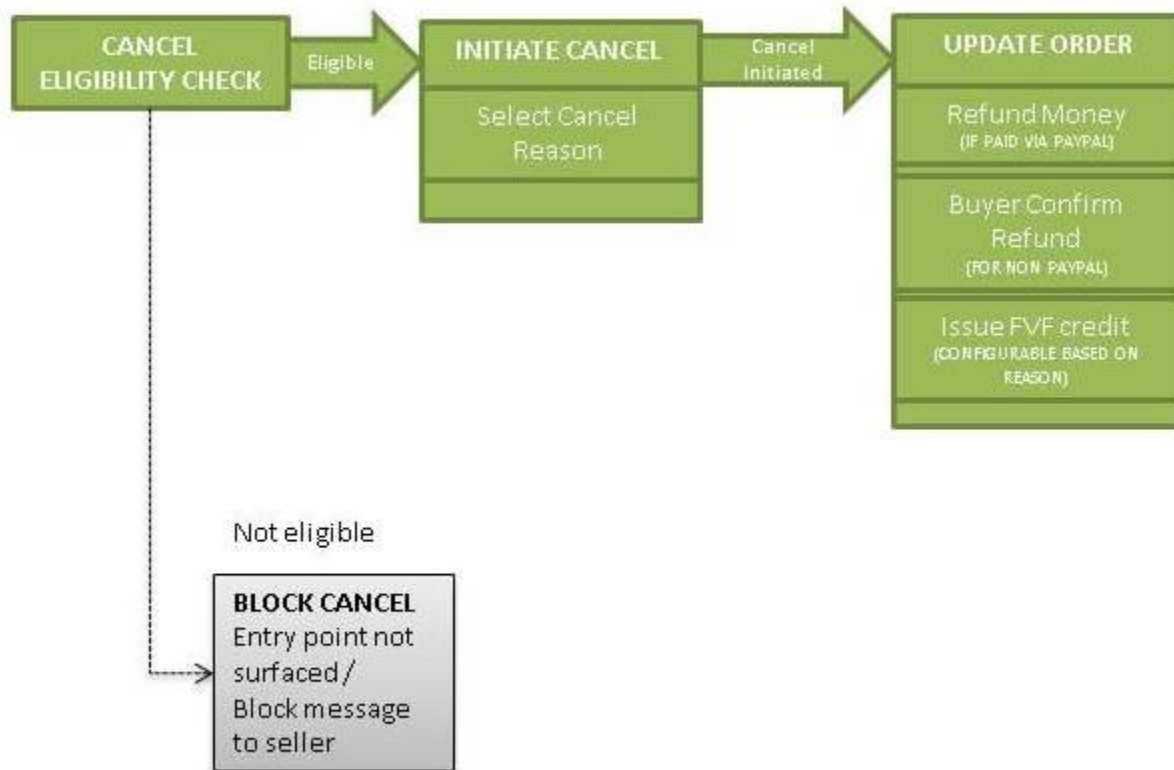


* Auto cancel configuration for sellers is planned for a future release. eBay’s auto-cancel policy is applicable for In Store Pickup and eBay Now purchases only.

A seller will be able to cancel an order using the new after-sale functionality. The granularity of the cancellation is the order level. Support for order line item cancellation is planned for a future release (timing TBD). The existing mutual cancellation feature triggered through the **AddDispute** call of the Trading API will still be supported. More details about mutual order cancellation are in the Integration Scenarios section.

Note that eBay Seller Standards will count an order cancellation for an “out-of-stock” reason as a seller defect, which may affect a seller’s Top Rated status and the benefits that status brings.

Figure 2 - Seller Initiated Order Cancellation



Order Cancellation-Supported Integration Scenarios

Please find the order cancellation scenario that is in line with your current integration to understand how changes in the APIs affect it. The label "existing" identifies an integration supported by an existing API (such as the Trading API); "existing modified" denotes either a change on the buyer's side or an existing API change. The label "new" means that the scenario described is based on the integration with new after-sale APIs and is recommended as a long-term solution based on the new APIs.

1. *[Existing] Buyer initiates cancellation:* a buyer contacts a seller using member-to-member messaging (or the **AddMemberMessageAAQToPartner** call in Trading API) and asks to cancel an (single line item) order or an order line item.

Seller actions:

- The seller creates a mutual cancellation using the **AddDispute** call of the Trading API.
- The seller issues a full refund using the PayPal API (or a bank/payment processor for other payment methods).

The order of actions is not relevant (the refund can be issued before creating the mutual cancellation).

Results of the actions:

- eBay will issue a Final Value Fee refund to the seller once the full refund is issued to the buyer.
- Seller standards will not interpret this scenario as an “out-of-stock” seller defect.

2. *[Existing modified] Buyer initiates cancellation:* a buyer submits a cancellation request using the new cancellation Web flow.

Seller actions:

- The seller learns about the cancellation request via email sent to the seller through the Web flow (or seller makes a **GetOrders** call and sees an **Order.CancelStatus** field in the response with a value of ‘CancelRequested’).
- The seller creates a mutual cancellation using the **AddDispute** call of the Trading API.
- The seller issues a full refund using the PayPal API (or a bank/payment processor for other payment methods).

The order of actions is not relevant (the refund can be issued before creating the mutual cancellation).

Results of the actions:

- eBay will close the buyer’s cancel request (in the new cancellation system) once the full refund is issued to the buyer.
- eBay will issue a Final Value Fee refund to the seller.
- Seller standards will not interpret this scenario as an “out-of-stock” seller defect.

Note: While creating a mutual cancellation is not mandatory, it ensures that the Final Value Fee is refunded to the seller and that the paid transaction/order status is marked as “cancelled,” even in the case when the seller issues a refund after the new cancellation system times out waiting for the seller’s approval (refer to the *Figure 1 – Buyer Initiated Cancellation* diagram).

3. *[Existing] Buyer initiates cancellation:* a buyer contacts a seller’s customer support and asks to cancel the (single line item) order or order line item (not using the new cancellation flow or eBay messaging).

Seller action:

- The seller issues a full refund using the PayPal API (or via bank/payment processor for other payment methods).

Important: Seller standards *will treat this scenario as an “out-of-stock” condition*. Sellers should avoid this scenario to prevent a seller defect.

4. *[New] Buyer initiates cancellation:* a buyer submits a cancellation request using the new cancellation Web flow.

Seller actions (assuming integration with the new Order Cancellation API):

- The seller learns about the cancellation request via email sent to the seller through the Web flow (or seller makes a **GetOrders** call and sees an **Order.CancelStatus** field in the response with a value of 'CancelRequested').
- The seller approves the cancellation request using the **/sell/order/v1/cancellation/approve** call of the Order Cancellation API.

Results of the actions:

- eBay will issue the full refund to the buyer if PayPal is the payment method; otherwise the seller is expected to issue a full refund (please refer to the *Figure 1* diagram).
- eBay will issue a Final Value Fee refund to the seller.
- Seller standards will not interpret this scenario as an “out-of-stock” seller defect.

5. *[New] Buyer initiates cancellation:* a buyer submits a cancellation request using the new cancellation Web flow.

Seller actions (assuming integration with the new Order Cancellation API):

- The seller learns about the cancellation request via email sent to the seller through the Web flow (or seller makes a **GetOrders** call and sees an **Order.CancelStatus** field in the response with a value of 'CancelRequested').
- The seller does nothing (no response to the request), or declines the cancellation request using the **/sell/order/v1/cancellation/reject** call of the Order Cancellation API.

Results of the actions:

- The seller is still expected to fulfill the order.
- eBay will send a notification to the buyer about the cancellation request being declined.

6. *[Existing] Seller initiates cancellation:*

Seller actions:

- The seller creates a mutual cancellation using the **AddDispute** call of the Trading API.
- The seller issues a full refund using the PayPal API (or a bank/payment processor for other payment methods).

The order of actions is not relevant (the refund can be issued before creating the mutual cancellation).

Result of the actions:

- Seller standards will not interpret this scenario as an “out-of-stock” seller defect if 'SellerRanOutOfStock' was the cancel explanation specified in the **AddDispute** call.

7. *[Existing] Seller initiates cancellation:*

Seller action:

- The seller issues a full refund using the PayPal API.

Result of the action:

- Seller standards will not interpret this scenario as an “out-of-stock” seller defect

8. *[New] Seller initiates cancellation .*

Seller action:

- The seller cancels an order using the **/sell/order/v1/cancellation/cancel** call of the Order Cancellation API.

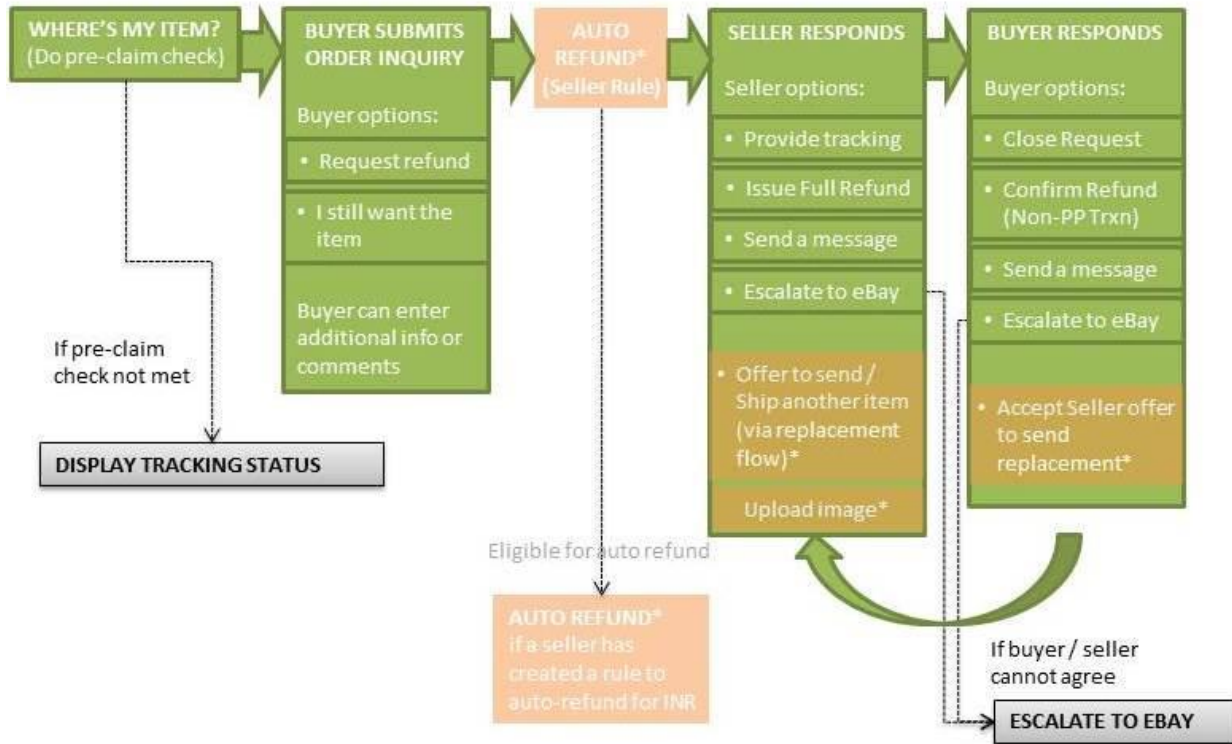
Results of the action:

- eBay will issue the full refund to the buyer if PayPal is the payment method; otherwise the seller is expected to issue a full refund (please refer to the *Figure 1* diagram).
- eBay will issue a Final Value Fee refund to the seller if the appropriate reason is specified.
- Seller standards will interpret this scenario as an “out-of-stock” seller defect if that was the specified reason.

Order Inquiry (INR – Item Not Received)

API changes summary: No changes to the Trading API’s **GetDispute**, **AddDispute**, **AddDisputeResponse**, **GetUserDisputes**, and **SellerReverseDispute** calls, nor to the existing Resolution Case Management API. The new Order Inquiry API is introduced for responding to or escalating an order inquiry.

Figure 3 - Order Inquiry diagram



* Auto-refund, sending a replacement and image upload are planned for a future release.

Seller standards will continue to count order inquiries (aka INR) towards the Seller Standards level.

Order Inquiry-Supported Integration Scenarios

Please find the order inquiry (INR) scenario that is in line with your current integration to understand how changes in the APIs affect it. The label "existing" identifies an integration supported by an existing API (such as the Trading API); "existing modified" denotes either a change on the buyer's side or an existing API change. The label "new" means that the scenario described is based on the integration with new after-sale APIs and is recommended as a long-term solution based on the new APIs.

1. *[Existing]* Buyer initiates an order inquiry about the shipping status: a buyer submits an inquiry using the Item Not Received flow; note that an eBay Buyer Protection INR case will be spawned by the inquiry so eBay can track buyer and seller actions and the final outcome.

Seller actions:

- The seller learns about the INR case through a **getUserCases** or **getEBPCaseDetail** call in the Resolution Case Management API.

- The seller responds to the buyer's inquiry on the eBay site, by using email, or member-to-member messaging.
2. *[Existing]* Buyer initiates an order inquiry about the shipping status: a buyer submits an inquiry using the INR flow; an INR case is created.

Seller actions:

- The seller learns about the INR case through a **getUserCases** or **getEBPCaseDetail** call in the Resolution Case Management API.
 - The seller responds to the inquiry via integration with the Resolution Case Management API to provide tracking (**provideTrackingInfo**), issue a refund (**issueFullRefund**), or escalate to Customer Service (**escalateToCustomerSupport**).
3. *[New]* Buyer initiates an order inquiry about the shipping status: a buyer submits an inquiry using the INR flow; an INR case is created.

Seller actions:

- The seller learns about the cancellation request via email sent to the seller through the Web flow (or seller makes a **GetOrders** call and sees an **Order.CancelStatus** field in the response with a value of 'CancelRequested').
- The seller learns about the INR case through the Trading API's **GetOrders** call, which returns an **InquiryStatus** value of 'TrackInquiryPendingSellerResponse' under the **Transaction.Status** container.
- The seller can first discover more details about the inquiry, and then respond to the buyer's inquiry by using the calls in the Order Inquiry API.

Returns and SNAD

API changes summary: The Trading API's **GetOrders** call now returns the new after-sale status fields, which includes returns; the Resolution Case Management API's **getUserCases** call is modified to include a Significantly Not as Described (SNAD) return (no interface change but a read logic changed to map SNAD return into SNAD case); the Return Management API is modified to include a new status for "return requested"; the new Order Return REST API is introduced.

The main change is that buyers will be routed to the returns flow if they have an issue with an item (aka SNAD).

Figure 4 - Seller accepts returns and the request is within the stated return window (US & UK)

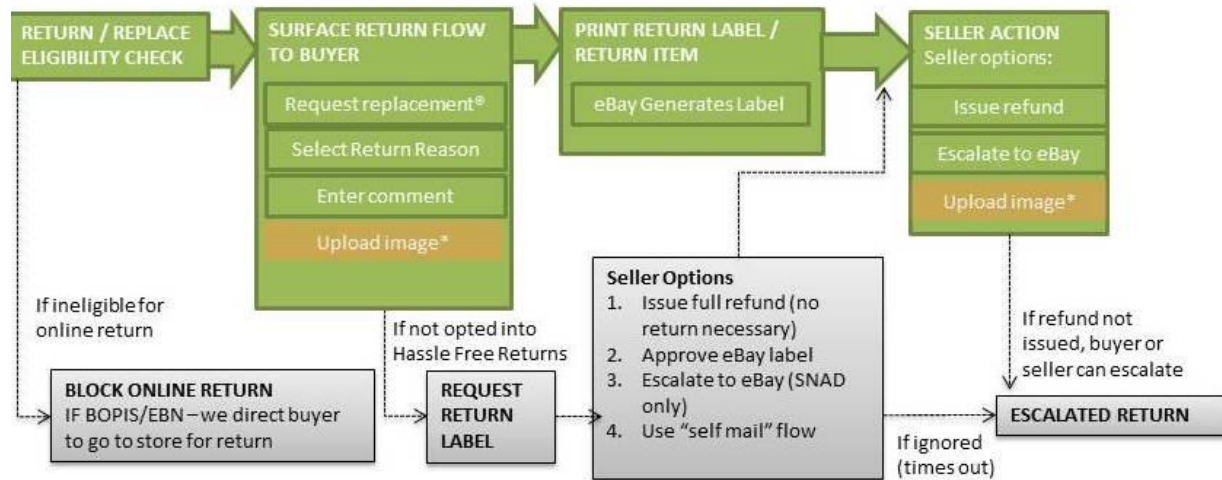


Figure 5 - Seller accepts returns and the request is within the stated return window (DE, AU & CA)

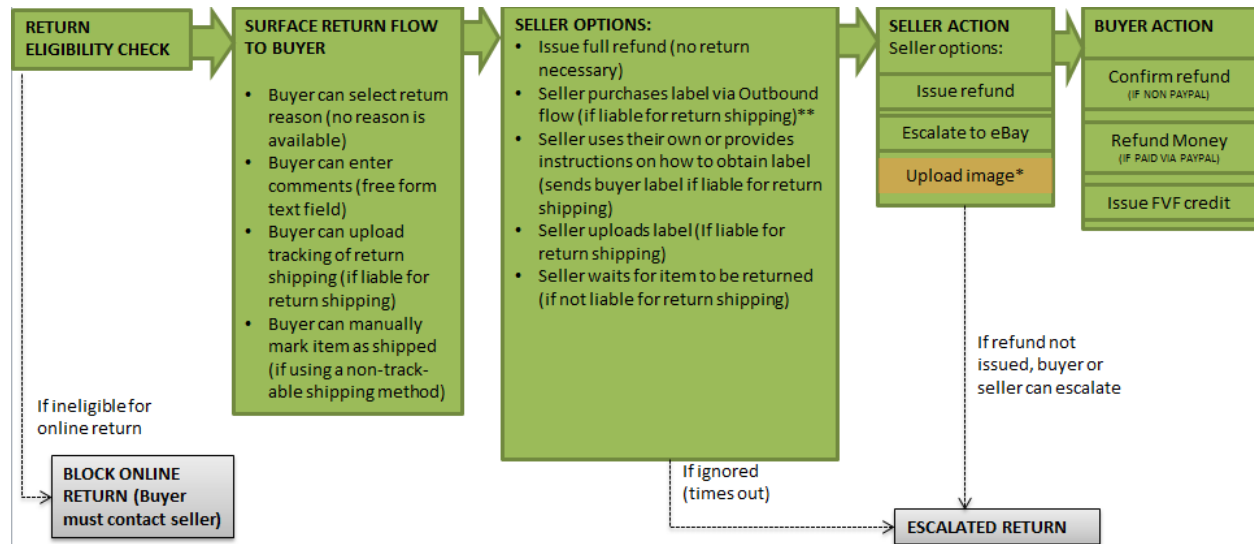
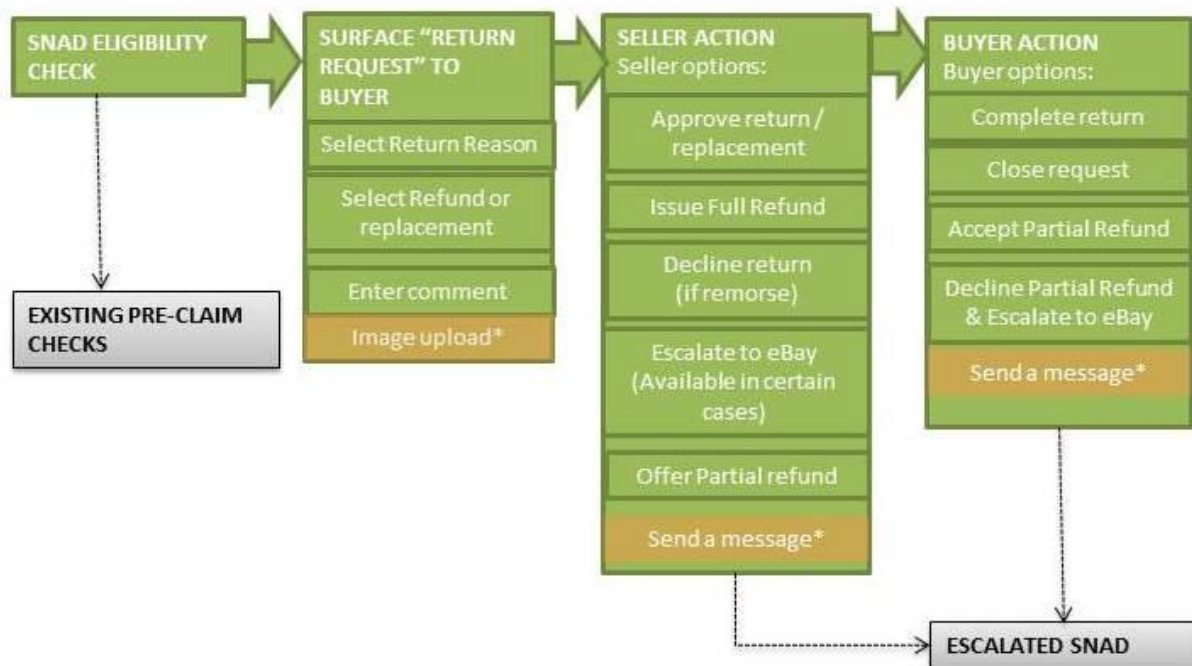


Figure 6 - Seller does not accept returns, or request is outside of the stated return window



Order Returns-Supported Integration Scenarios

Please find the scenario that best describes your current integration to understand how changes in the APIs affect it. The label "existing" identifies an integration supported by an existing API (such as the Trading API); "existing modified" denotes either a change on the buyer's side or an existing API change. The label "new" means that the scenario described is based on the integration with new after-sale APIs and is recommended as a long-term solution based on the new APIs.

1. *[Existing modified]* Buyer wants to return an item for a SNAD reason, following eBay's and seller's return policy. This results in buyer submitting a request for a return: previously a buyer would have been routed to the Resolution Center to create a SNAD case. With the product change, a buyer can submit the request in the returns flow and a return request will be created with the new "return requested" status.

Seller actions:

Seller actions:

- The seller learns about the return case in one of the following ways:
 - i. By using **GetOrders** in the Trading API. If a return request for an order line item was initiated by the buyer, a **ReturnStatus** value of 'ReturnRequestPendingApproval' will be returned under the **Transaction.Status** container.

- ii. By using **getUserReturns** in the Return Management API. A new value returned in the **returns.status** to indicated that a return request has been made.
 - iii. By using **getUserCases** in the Resolution Case Management API. No change to the interface.
- The seller responds to the request on the eBay site, or by using email or member-to-member messaging.

CAPABILITY GAP: The legacy Resolution Case Management and Return Management APIs will NOT be able to approve or decline a request for return; the options available are as described in scenarios 1 and 2.

2. *[New] Buyer wants to return an item for a SNAD reason, following eBay's and seller's stated return policies; this results in buyer submitting a request for return [same as scenario 1]:* A return case is created with the new "return requested" status.

Seller actions:

- The seller learns about the return request by using **GetOrders** in the Trading API. If a return request for an order line item was initiated by the buyer, a **ReturnStatus** value of 'ReturnRequestPendingApproval' will be returned under the **Transaction.Status** container.
- The seller can first discover more details about the return request, and then respond to the buyer's return request by using the calls in the Order Return API.

3. *[Existing modified] Buyer wants to return an item following eBay's and seller's stated return policies; this results in buyer submitting a return (no seller's approval needed):* A return case is created with a status of "ready for shipping."

Seller's actions:

- The seller learns about the return request in one of the following ways:
 - i. Using **GetOrders** in the Trading API [existing modified]. If a return request for an order line item was initiated by the buyer, a **ReturnStatus** value of 'ReturnRequestPendingApproval' will be returned under the **Transaction.Status** container.
 - ii. Using **getUserReturns** in the Return Management API [existing]. A return ID is returned in the response.
- The seller manages the return using the Return Management API or on the eBay site (a refund issued on the eBay site provides info such as return address or tracking for a replacement).

4. *[New] Buyer wants to return an item following eBay's and seller's stated return policies; this results in buyer submitting a return (no seller's approval needed) [same as scenario 3]:* A return request is created with a status of "ready for shipping."

Seller's actions:

- The seller learns about the return case using **GetOrders** in the Trading API [existing modified]. If a return request for an order line item was initiated by the buyer, a

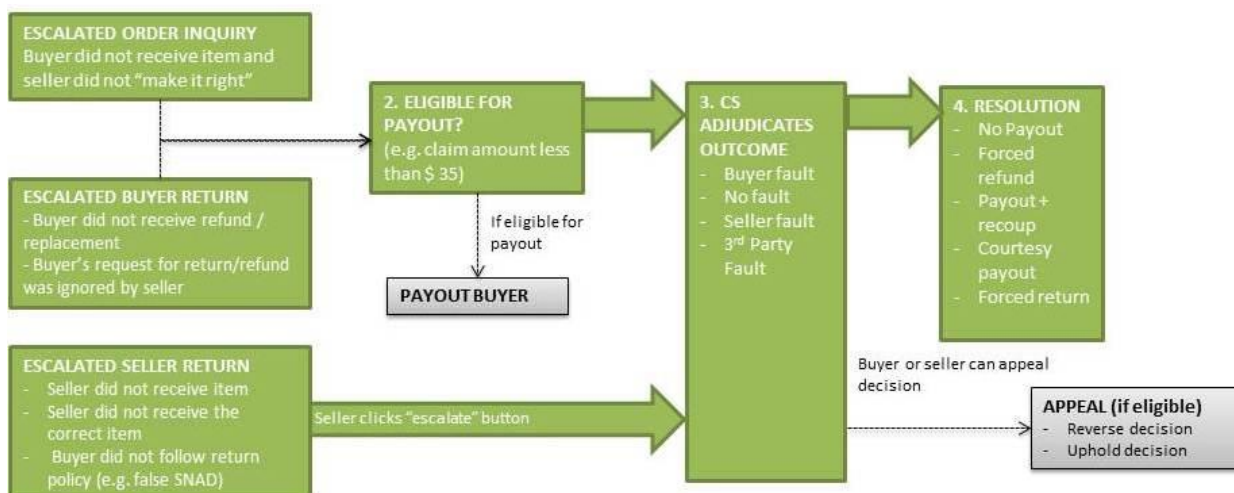
ReturnStatus value of 'ReturnRequestPendingApproval' will be returned under the **Transaction.Status** container.

- The seller gets the details of the return case through the new Order Return REST API.
- The seller approves the request and manages the return via integration with the new Order Return REST API.

Escalations

API changes summary: If the buyer and seller are not able to come to a satisfactory agreement during after-sale self-service flows, there will be an option to escalate to eBay Customer Support, which will adjudicate and determine an outcome. The eBay Buyer Protection program will continue to govern the policy determining when buyer can escalate and how Customer Service adjudicates cases. Sellers will also be able to escalate an order inquiry or a return.

Figure 7 - Buyer and Seller Escalation to Customer Service After Sale



Escalations Supported Integration Scenario(s)

Please find the scenario that best describes your current integration to understand how changes in the APIs affect it. The label "existing" identifies an integration supported by an existing API (such as the Trading API); "existing modified" denotes either a change on the buyer's side or an existing API change. The label "new" means that the scenario described is based on the integration with new after-sale APIs and is recommended as a long-term solution based on the new APIs.

1. *[Existing modified] Buyer has escalated an order inquiry (INR).*

Seller's actions:

- The seller learns about the escalated INR case: through the Trading API's **GetOrders** call, which returns an **InquiryStatus** value of 'TrackInquiryEscalatedPendingSeller' under the **Transaction.Status** container; OR through Resolution Case Management API's **getUserCases** or **getEBPCaseDetail** call
- The seller manages the escalated INR case (issues refund, provides shipping and tracking info) using the Resolution Case Management API [as is], or on the eBay site.

2. *[Existing modified] Buyer has escalated a return request.*

Seller's actions:

- The seller learns about the escalated return request: through the Trading API's **GetOrders** call, which returns an **ReturnStatus** value of 'ReturnEscalatedPendingSeller' under the **Transaction.Status** container; OR discovered as a SNAD case through Resolution Case Management API's **getUserCases** or **getEBPCaseDetail** call; OR discovered as a escalated return case through Return Management API's **getUserReturns** call.
- The seller manages the escalation (issue refund, provide shipping and tracking info) using the Resolution Case Management API [as is], or on the eBay site.

3. *[Existing] Seller wants to escalate an order inquiry (INR).*

- The seller can escalate using the Resolution Case Management API's **escalateToCustomerSupport** call [as is].
- The seller manages escalation using the Resolution Case Management API [as is].

4. *[New] Seller wants to escalate an order inquiry (INR).*

- The seller can escalate using the **/sell/order/v1/inquiry/escalation** call of the new Order Inquiry API.
- The seller also manages the escalation using the new Order Resolution API.

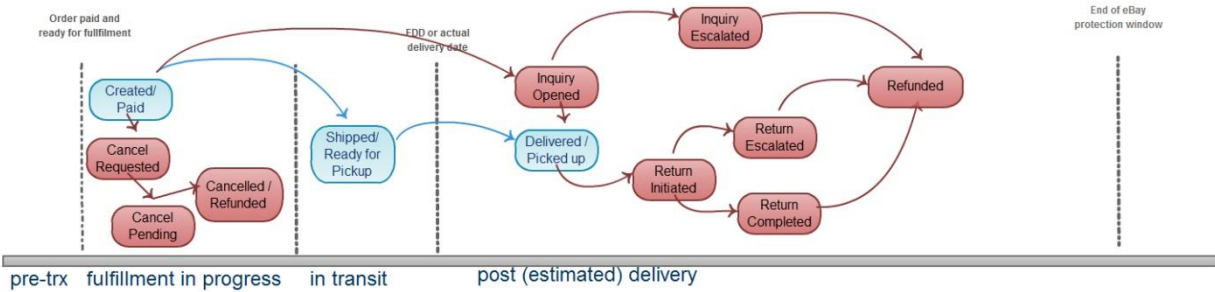
5. *[New] Seller wants to escalate a return or return request.*

- The seller can escalate using the **/sell/order/v1/return/{returnId}/escalate** call of the new Order Return API.
- The seller manages the escalation using the new Order Resolution API.

CAPABILITY GAP: The legacy Resolution Case Management and Return Management APIs are NOT able to escalate a return or a request for return, or manage such escalations; but the new after-sale APIs have that capability.

New After-Sale API Set

Order in Post Transaction Lifecycle



Making a Call

This section provides the general information you need to make a call with the new after-sale APIs.

Root Endpoints

Each API call has a unique endpoint, but all calls share common root endpoints for the supported environments:

Sandbox root endpoint:

<https://api.sandbox.ebay.com>

Production root endpoint:

<https://api.ebay.com>

Note: When making calls, be sure to use an authorization token that is generated for the correct environment.

Supported Formats

The new *after-sale* APIs are RESTful APIs that support JSON and XML request and response bodies. See the HTTP Headers section for details on specifying the desired format.

HTTP Headers

The following table lists the HTTP headers supported by the after-sale APIs.

Header	Description	Applicable Values	Required?
Accept	The media types for the response. Defaults to JSON.	application/xml, application/json	No

Header	Description	Applicable Values	Required?
Authorization	The Authentication Token for the user upon whose behalf the call is made. For more detailed information on getting and renewing authentication tokens, see Getting Tokens .	A valid eBay authorization token, prefixed by 'TOKEN' (TOKEN <authorization token value>)	Yes
Content-Type	The MIME type (used with POST requests) of the body of the request. Defaults to JSON.	application/xml, application/json	Yes
X-EBAY-C-MARKETPLACE-ID	The global ID of an eBay site.	For example: EBAY-US.	Yes

Order Cancellation API

For a representation of the full payload structure and content of each call, see the Order Cancellation API skeleton documentation.

POST /buy/order/v1/cancellation/cancel

Cancel an order on behalf of the buyer.

Note: The `POST /sell/order/v1/cancellation/cancel` call provides the same functionality for the seller.

Sample request

```
https://api.ebay.com/buy/order/v1/cancellation/cancel
{
  "orderId": "170006489164-8753917007!5585902017"
}
```

Input Container/Field	Type	Occurrence	Available Values
buyerPaid	boolean	Conditional	
buyerPaidDate	dateTime	Conditional	

Input Container/Field	Type	Occurrence	Available Values
cancelReason	cancelReasonEnum	Optional	BUYER_CANCEL_OR_ADDRESS_ISSUE FOUND_CHEAPER_PRICE ORDER_MISTAKE OTHER OUT_OF_STOCK_OR_CANNOT_FULFILL PRICE_TOO_HIGH WONT_ARRIVE_IN_TIME
invoiceId	long	Conditional	
itemId	long	Conditional	
orderId	string	Conditional	
transactionId	long	Conditional	

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "cancelId": 5000001070
}
```

GET /buy/order/v1/cancellation/checkEligibility

Check the eligibility of the order to be cancelled on behalf of the buyer.

Note: The `GET /sell/order/v1/cancellation/checkEligibility` call provides the same functionality for the seller.

Sample request

```
https://api.ebay.com/buy/order/v1/cancellation/checkEligibility?orderId=170006559499-8795427007!5614515017
```

Query Parameter	Type	Occurrence
invoiceId	string	Conditional

itemId	string	Conditional
orderId	string	Conditional
transactionId	string	Conditional

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "orderId": "170006559499-8795427007!5614515017",
  "eligible": true,
  "failureReason": [],
  "itemEligibilityResult": [
    {
      "itemId": 170006559499,
      "transactionId": 8795427007,
      "eligible": true,
      "failureReason": []
    }
  ]
}
```

POST /buy/order/v1/cancellation/confirm

Confirm the buyer's request to cancel an order.

Sample request

```
https://api.ebay.com/buy/order/v1/cancellation/confirm
{
  "notPaid": "true",
  "cancelId": "5000001080"
}
```

Input Container/Field	Type	Occurrence
cancelId	long	Required
notPaid	boolean	Required
refundDate	dateTime	Optional

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200
}
```

GET /buy/order/v1/cancellation/find

Find an order based on specified criteria from the buyer.

Note: The `GET /sell/order/v1/cancellation/find` call provides the same functionality for the seller.

Sample request

```
https://api.ebay.com/buy/order/v1/cancellation/find?itemId=170006475517&transactionId=8749233007
```

Query Parameter	Type	Occurrence
<code>cancelId</code>	string	Conditional
<code>creationDateRangeFrom</code>	string	Optional
<code>creationDateRangeTo</code>	string	Optional
<code>entriesPerPage</code>	string	Optional
<code>invoiceId</code>	string	Conditional
<code>itemId</code>	string	Conditional
<code>orderId</code>	string	Conditional
<code>pageNumber</code>	string	Optional
<code>sellerLoginName</code>	string	Optional
<code>sort</code>	string	Optional
<code>transactionId</code>	string	Conditional

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "cancellations": [
    {
      "cancelId": 5000000532,
      "marketplaceId": "EBAY_US",
      "orderId": "170006475517-8749233007!5582554017",
      "requestorType": "SELLER",
    }
  ]
}
```

```

        "cancelReason": "BUYER_CANCEL_OR_ADDRESS_ISSUE",
        "state": "CLOSED",
        "status": "CANCEL_PENDING",
        "closeReason": "FULL_REFUNDED",
        "paymentStatus": "PAYPAL_PAID",
        "requestRefundAmount": {
            "value": 1.23,
            "currency": "USD"
        },
        "cancelRequestDate": {
            "value": "2014-05-31T00:07:53.000Z",
            "formattedValue": "2014-05-31T00:07:53.000Z"
        }
    }
},
"total": 1,
"paginationOutput": {
    "pageNumber": 1,
    "entriesPerPage": 1,
    "totalPages": 1,
    "totalEntries": 1,
    "any": null
}
}

```

GET /buy/order/v1/cancellation/{cancelId}

Retrieve the details of an order cancellation for the buyer.

Note: The `GET /sell/order/v1/cancellation/{cancelId}` call provides the same functionality for the seller.

Sample request

```
https://api.ebay.com/buy/order/v1/cancellation/5000001071
```

Path Parameter	Type	Occurrence
cancelId	long	Required

Sample Response

```

{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "cancelDetail": {
    "cancelId": 5000001071,
    "marketplaceId": "EBAY_US",
    "orderId": "170006494376-8756205007!5585908017",
    "requestorType": "SELLER",
    "cancelReason": "BUYER_CANCEL_OR_ADDRESS_ISSUE",
    "state": "REFUND_PENDING",
    "status": "CANCEL_PENDING",
    "paymentStatus": "PAYPAL_PAID",

```

```

    "requestRefundAmount": {
      "value": 2.46,
      "currency": "USD"
    },
    "cancelRequestDate": {
      "value": "2014-08-22T23:29:11.000Z",
      "formattedValue": "2014-08-22T23:29:11.000Z"
    },
    "lineItems": [
      {
        "itemId": 170006494376,
        "transactionId": 8756205007
      }
    ],
    "activityHistories": [
      {
        "activityType": "SELLER_CREATE_CANCEL",
        "activityParty": "SELLER",
        "actionDate": {
          "value": "2014-08-22T23:29:07.000Z",
          "formattedValue": "2014-08-22T23:29:07.000Z"
        },
        "stateFrom": "INITIAL",
        "stateTo": "REFUND_PENDING"
      }
    ]
  }
}

```

POST /sell/order/v1/cancellation/approve

Approve a requested order cancellation on behalf of the seller.

Sample request

```

https://api.ebay.com/sell/order/v1/cancellation/approve
{
  "cancelId": "5000001070"
}

```

Input Container/Field	Type	Occurrence
cancelId	long	Required

Sample Response

```

{
  "ackValue": "SUCCESS",
  "responseStatus": 200
}

```

POST /sell/order/v1/cancellation/cancel

Cancel an order on behalf of the seller.

Note: The `POST /buy/order/v1/cancellation/cancel` call provides the same functionality for the buyer.

Sample request

```
https://api.ebay.com/sell/order/v1/cancellation/cancel
{
  "orderId": "170006494376-8756205007!5585908017",
  "cancelReason": "BUYER_CANCEL_OR_ADDRESS_ISSUE"
}
```

Input Container/Field	Type	Occurrence	Available Values
cancelReason	cancelReasonEnum	Required	BUYER_CANCEL_OR_ADDRESS_ISSUE FOUND_CHEAPER_PRICE ORDER_MISTAKE OTHER OUT_OF_STOCK_OR_CANNOT_FULFILL PRICE_TOO_HIGH WONT_ARRIVE_IN_TIME
invoiceId	long	Conditional	
itemId	long	Conditional	
orderId	string	Conditional	
transactionId	long	Conditional	

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "cancelId": 5000001071
}
```


GET /sell/order/v1/cancellation/checkEligibility

Check the eligibility of the order to be cancelled on behalf of the seller.

Note: The `GET /buy/order/v1/cancellation/checkEligibility` call provides the same functionality for the buyer.

Sample request

```
https://api.ebay.com/sell/order/v1/cancellation/checkEligibility?orderId=170006559499-8795427007!5614515017
```

Query Parameter	Type	Occurrence
invoiceId	string	Conditional
itemId	string	Conditional
orderId	string	Conditional
transactionId	string	Conditional

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "orderId": "170006559499-8795427007!5614515017",
  "eligible": true,
  "failureReason": [],
  "itemEligibilityResult": [
    {
      "itemId": 170006559499,
      "transactionId": 8795427007,
      "eligible": true,
      "failureReason": []
    }
  ]
}
```

GET /sell/order/v1/cancellation/find

Find an order based on specified criteria from the seller.

Note: The `GET /buy/order/v1/cancellation/find` call provides the same functionality for the buyer.

Sample request

```
https://api.ebay.com/sell/order/v1/cancellation/find?itemId=170006475517&transactionId=8749233007
```

Query Parameter	Type	Occurrence
cancelId	string	Conditional
creationDateRangeFrom	string	Optional
creationDateRangeTo	string	Optional
entriesPerPage	string	Optional
invoiceId	string	Conditional
itemId	string	Conditional
orderId	string	Conditional
pageNumber	string	Optional
buyerLoginName	string	Optional
sort	string	Optional
transactionId	string	Conditional

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "cancellations": [
    {
      "cancelId": 5000000532,
      "marketplaceId": "EBAY_US",
      "orderId": "170006475517-8749233007!5582554017",
      "requestorType": "SELLER",
      "cancelReason": "BUYER_CANCEL_OR_ADDRESS_ISSUE",
      "state": "CLOSED",
      "status": "CANCEL_PENDING",
      "closeReason": "FULL_REFUNDED",
      "paymentStatus": "PAYPAL_PAID",
      "requestRefundAmount": {
        "value": 1.23,
        "currency": "USD"
      },
      "cancelRequestDate": {
        "value": "2014-05-31T00:07:53.000Z",
        "formattedValue": "2014-05-31T00:07:53.000Z"
      }
    }
  ]
}
```

```

    }
  },
  "total": 1,
  "paginationOutput": {
    "pageNumber": 1,
    "entriesPerPage": 1,
    "totalPages": 1,
    "totalEntries": 1,
    "any": null
  }
}

```

POST /sell/order/v1/cancellation/reject

Reject a requested order cancellation on behalf of the seller.

Sample request

```

https://api.ebay.com/sell/order/v1/cancellation/reject
{
  "cancelId": "5000001083",
  "shipmentDate": {
    "value": "2014-08-22T23:29:11.000Z"
  },
  "trackingNumber": "4209408792748901015477100002652680"
}

```

Input Container/Field	Type	Occurrence
cancelId	long	Required
shipmentDate	dateTime	Optional
trackingNumber	string	Optional

Sample Response

```

{
  "ackValue": "SUCCESS",
  "responseStatus": 200
}

```

GET /sell/order/v1/cancellation/{cancelId}

Retrieve the details of an order cancellation for the seller.

Note: The `GET /buy/order/v1/cancellation/{cancelId}` call provides the same functionality for the buyer.

Sample request

```
https://api.ebay.com/sell/order/v1/cancellation/5000001071
```

Path Parameter	Type	Occurrence
cancelId	long	Required

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "cancelDetail": {
    "cancelId": 5000001071,
    "marketplaceId": "EBAY_US",
    "orderId": "170006494376-8756205007!5585908017",
    "requestorType": "SELLER",
    "cancelReason": "BUYER_CANCEL_OR_ADDRESS_ISSUE",
    "state": "REFUND_PENDING",
    "status": "CANCEL_PENDING",
    "paymentStatus": "PAYPAL_PAID",
    "requestRefundAmount": {
      "value": 2.46,
      "currency": "USD"
    },
    "cancelRequestDate": {
      "value": "2014-08-22T23:29:11.000Z",
      "formattedValue": "2014-08-22T23:29:11.000Z"
    },
    "lineItems": [
      {
        "itemId": 170006494376,
        "transactionId": 8756205007
      }
    ],
    "activityHistories": [
      {
        "activityType": "SELLER_CREATE_CANCEL",
        "activityParty": "SELLER",
        "actionDate": {
          "value": "2014-08-22T23:29:07.000Z",
          "formattedValue": "2014-08-22T23:29:07.000Z"
        },
        "stateFrom": "INITIAL",
        "stateTo": "REFUND_PENDING"
      }
    ]
  }
}
```

Order Inquiry API

POST /buy/order/v1/inquiry

Create an INR (item not received) request for a specified order and item on behalf of the buyer.

Sample Request

```
https://api.ebay.com/buy/order/v1/inquiry
{
  "orderId":8149458,
  "transactionId": 8378946011,
  "itemId": 350004393501,
  "desiredOutcome": "FULL_REFUND",
  "message": "I did not receive the item. Please provide me a refund.",
  "claimQuantity":1
}
```

Input Container/Field	Type	Occurrence	Description
desiredOutcome	DesiredOutcomeTypeEnum	Required	Available values: FULL_REFUND STILL_WANT_ITEM

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the request has been fulfilled and the INR has been created).

POST /buy/order/v1/inquiry/closure

Close an inquiry on behalf of the buyer.

Sample Request

```
https://api.ebay.com/buy/order/v1/inquiry/closure
{
  "inquiryId":5002619698,
  "buyerCloseReason":"itemArrived",
  "buyerComments":"Seller added shipment info"
}
```

Input Container/Field	Type	Occurrence
-----------------------	------	------------

Input Container/Field	Type	Occurrence
buyerCloseReason	BuyerCloseReason	Available values: itemArrived workedOutWithSeller wouldRatherKeepTheItem other

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 200.

Sample Error Response

The buyer did not provide a valid reason to close the request:

```
{
  errorMessage: {
    error: [
      domain: "resolutions",
      errorId: 292039,
      category: "Request",
      severity: "Error",
      message: "A valid close reason is required.",
      longMessage: "Please supply a valid close reason to process your
request.",
      actionToTake: CorrectInputData
    ]
  }
  extension: [
  ]
  ackValue : "FAILURE",
  responseStatus: 400,
  message: null
}
```

POST /buy/order/v1/inquiry/eligibility

Determine if the buyer is eligible to file an inquiry for a particular order and item. Various criteria must be satisfied, including the timing of the inquiry, the eBay site involved, and approval of eligibility by PayPal.

Sample Request

```
https://api.ebay.com/buy/order/v1/inquiry/eligibility
{
  "orderId": 17087987,
  "transactionId": 9104438007,
```

```

"itemId": 170007164603,
"site": "EBAY_US",
"checkType": "PRE_CLAIM_CHECK"
}

```

Input Container/Field	Type	Occurrence	Description
checkType	EligibilityCheckTypeEnum	Required	Type of eligibility check. Available values: CLAIM_PENDING_CHECK CS_ESCALATION_CHECK POST_CLAIM_CHECK PRE_CLAIM_CHECK RETURNS_PRE_CLAIM_CHECK
site	SiteTypeEnum	Required	eBay site to which this call applies. Available values: EBAY_AU Australia EBAY_CA Canada EBAY_DE Germany EBAY_MAIN main ebay US site EBAY_UK United Kingdom EBAY_US alias for EBAY_MAIN

Sample Response

```

{
  "eligibleToFileRequest": false,
  "latestPossibleFilingDate": {
    "value": "2014-10-05T06:59:59.000Z",
    "formattedValue": "2014-10-05T06:59:59.000Z"
  },
}

```

```

    "earliestPossibleFilingDate": {
      "value": "2014-09-05T07:00:00.000Z",
      "formattedValue": "2014-09-05T07:00:00.000Z"
    },
    "estimatedDeliveryDate": {
      "value": "2014-09-04T07:00:00.000Z",
      "formattedValue": "2014-09-04T07:00:00.000Z"
    },
    "paypalReasonToDenyRequestFiling": null,
    "warnings": null,
    "errors": [{
      "id": "FILING_TOO_SOON",
      "severity": "ERROR",
      "parameters": null
    }],
    "nonOverridableErrors": null,
    "eligibleWithWarning": false
  }
}

```

Output Container/Field	Type	Description
paymentProcessorReasonToDenyRequestFiling	PaymentDenialReasonEnum	Reason for payment denial. Available values: PP_ACH_RETURN PP_ADMINISTRATIVE_FRAUD_REVIEW PP_CREDITCARD_CHARGEBACK PP_DISALLOWED_DISPUTE PP_ELV_REVERSAL PP_PAYPAL_GUEST PP_UNAUTHORIZED_TRANSACTION

Request (client side) Errors

Errors indicating client side (request related) issues use HTTP status 4xx.

- Status 400 is used for errors in the request, such as invalid / missing query parameters, bad HTTP header values, etc.
- Status 404 is used for missing/incorrect resources in the URI path of a RESTful query.
- Status 409 is used in situations where the request (such as a POST/PUT request) is not consistent with current object state.

Invalid orderId:

```
{
  errorMessage: {
    error: [
      domain: "resolutions",
      errorId: 292001,
      category: "Request",
      severity: "Error",
      message: "A valid order ID is required.",
      longMessage: "Please supply a valid order ID to process your request.",
      actionToTake: CorrectInputData
    ]
  }
  extension: [
  ]
  ackValue : "FAILURE",
  responseStatus: 400,
  message: null
}
```

Bad HTTP header values:

```
{
  errorMessage: {
    error: [
      domain: "resolutions",
      errorId: 292004,
      category: "Request",
      severity: "Error",
      message: "A valid HTTP header is required.",
      longMessage: "The HTTP headers contained in the request appeared to be
invalid. Please verify that the headers in your request are valid",
      actionToTake: CorrectInputData
    ]
  }
  extension: [
  ]
  ackValue : "FAILURE",
  responseStatus: 400,
  message: null
}
```

Application (service) errors

These errors use HTTP status 500. Examples:

- The service catches an exception in its business logic.
- A resource (database, underlying service) needed by the service is unavailable.

transactionId not found:

```
{
  errorMessage: {
    error: [
      domain: "resolutions",
```

```

        errorId: 292008,
        category: "Application",
        severity: "Error",
        message: "No such transaction ID '%1'.",
        longMessage: "No such transaction ID '%1'. Please provide a valid
transaction ID.",
        actionToTake: CorrectInputData,
        parameters: [
            <TXN_ID AS THE FIRST ELEMENT IN THIS ARRAY>
        ]
    ]
}
extension: [
]
ackValue : "FAILURE",
responseStatus: 500,
message: null
}

```

Check eligibility service catches an exception in its business logic:

```

{
    errorMessage: {
        error: [
            domain: "resolutions",
            errorId: 292010,
            category: "Application",
            severity: "Error",
            message: "application error '%1'",
            longMessage: "Encountered error '%1'. If this failure continues, contact
support.",
            actionToTake: Retry,
            parameters: [
                <THE NAME OF THE APPLICATION ERROR AS THE FIRST ELEMENT IN THIS ARRAY>
            ]
        ]
    }
    extension: [
    ]
    ackValue : "FAILURE",
    responseStatus: 500,
    message: null
}

```

Database unavailable:

```

{
    errorMessage: {
        error: [
            domain: "resolutions",
            errorId: 292011,
            category: "Application",
            severity: "Error",
            message: "DB connection unavailable.",
            longMessage: "Unable to connect to database '%1'. If this failure
continues, contact DB support.",
            actionToTake: Retry,
            parameters: [

```

```
        <DB NAME AS THE FIRST ELEMENT IN THIS ARRAY>
    ]
}
extension: [
]
ackValue : "FAILURE",
responseStatus: 500,
message: null
}
```

Service unavailable:

```
{
  errorMessage: {
    error: [
      domain: "resolutions",
      errorId: 292012,
      category: "Application",
      severity: "Error",
      message: "Service connection unavailable.",
      longMessage: "Unable to connect to service '%1'. If this failure
continues, please contact support.",
      actionToTake: Retry,
      parameters: [
        <SERVICE NAME AS THE FIRST ELEMENT IN THIS ARRAY>
      ]
    ]
  }
  extension: [
  ]
  ackValue : "FAILURE",
  responseStatus: 500,
  message: null
}
```

System Errors (errors in logic outside of the service application logic itself)

These are server side errors in framework / container processing logic that is outside the service; they use status 500. Examples:

- Service framework cannot identify service to dispatch to or has some problem in dispatching.
- Service processed the request, but postprocessing failed in the service framework. In this case there may be both Service and System Errors returned in the response.

Framework: Service dispatch issues:

```
{
  errorMessage: {
    error: [
      domain: "resolutions",
      errorId: 292013,
      category: "System",
      severity: "Error",
      message: "Framework unable to delegate to a service.",

```

```

        longMessage: "The service framework was unable to identify a service to
dispatch to or had some trouble in dispatching the request.",
        actionToTake: ConsultTechSupport
    ]
}
extension: [
]
ackValue : "FAILURE",
responseStatus: 500,
message: null
}

```

Framework: Postprocessing failed in framework issues:

```

{
  errorMessage: {
    error: [
      domain: "resolutions",
      errorId: 292014,
      category: "System",
      severity: "Error",
      message: "Postprocessing failed in framework",
      longMessage: "The service processed the request, but subsequent processing
failed at the framework level.",
      actionToTake: ConsultTechSupport,
      inputRefId: "<JSON OF THE REQUEST THAT FAILED>"
    ]
  }
  extension: [
  ]
  ackValue : "FAILURE",
  responseStatus: 500,
  message: null
}

```

POST /buy/order/v1/inquiry/escalation

Escalate the specified inquiry for the buyer.

Note: The `POST /sell/order/v1/inquiry/escalation` call provides the same functionality for the seller.

Sample Request

```

https://api.ebay.com/buy/order/v1/inquiry/escalation
{
  "escalationReason": "OTHERS",
  "message": "Please help",
  "inquiryId": 5002657723
}

```

Input Container/Field	Type	Occurrence	Description
-----------------------	------	------------	-------------

Input Container/Field	Type	Occurrence	Description
escalationReason	ContactCSEnum	Required	Basis for escalation. Available values: SELLER_NORESPONSE TROUBLE_COMMUNICATION NOT_RECEIVED OTHERS

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the inquiry has been escalated).

POST /buy/order/v1/inquiry/message

Send a message from the buyer regarding the specified inquiry.

Note: The `POST /sell/order/v1/inquiry/message` call provides the same functionality for the seller.

Sample Request

```
https://api.ebay.com/buy/order/v1/inquiry/message
{
  "inquiryId":5002661989,
  "message": "Please refund my money"
}
```

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the message has been sent).

POST /buy/order/v1/inquiry/refundConfirmation

With this call the buyer confirms receipt of a full refund for the specified inquiry.

Sample Request

```
https://api.ebay.com/buy/order/v1/inquiry/refundConfirmation
{
```

```
}
  "inquiryId": 5002619614
}
```

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the confirmation has been generated and sent).

GET /buy/order/v1/inquiry/{inquiryId}

Retrieve details about a specified inquiry for the buyer.

Note: The `GET /sell/order/v1/inquiry/{inquiryId}` call provides the same functionality for the seller.

Sample Request

This call has no request payload.

```
https://api.ebay.com/buy/order/v1/inquiry/5002661989
```

Sample Response

```
{
  "inquiryId": 5002661989,
  "orderId": 0,
  "itemId": 350004393501,
  "transactionId": 8378946011,
  "buyer": "ebp_us_buyer_140554539108450",
  "seller": "ebp_us_seller_140554537847515",
  "state": "pendCreated",
  "status": "inactive",
  "claimAmount": {
    "value": 101.0,
    "currency": "USD"
  },
  "shippingFee": {
    "value": 0.0,
    "currency": "USD"
  },
  "transactionSiteId": 0,
  "inquiryQuantity": 1,
  "initiator": "buyer",
  "smirDate": {
    "value": "2014-09-08T21:16:50.000Z",
    "formattedValue": "2014-09-08T21:16:50.000Z"
  },
  "viewPPTrasactionUrl":
"http://payments.qa.ebay.com/ws/eBayISAPI.dll?UnifiedCheckoutSummaryRedirect2PP&buyero
rseller=0&itemid=350004393501&transid=8378946011",
  "sellerClosureReason": "undefined",
  "buyerClosureReason": "undefined",
  "eCheckClearanceDate": null,
}
```

```

"lastEscalationDateforECheck": null,
"ebpEligible": false,
"itemDetails": {
  "itemTitle": "Multi item Single seller Order PPWed Jul 16 14:16:41 MST 2014",
  "itemPrice": {
    "value": 100.0,
    "currency": "USD"
  },
  "itemPictureUrl":
"http://i.ebayimg.qa.ebay.com/00/s/MTA2NlgxNjAw/z/2oEAANKvLslTxuvE/$_1.JPG?set_id=2",
  "viewPurchasedItemUrl":
"http://cgi.qa.ebay.com/ws/eBayISAPI.dll?ViewItemVersion&item=350004393501&tid=8378946011&view=all"
},
"inquiryDetails": {
  "creationDate": {
    "value": "2014-09-04T21:16:50.000Z",
    "formattedValue": "2014-09-04T21:16:50.000Z"
  },
  "holdEndDate": null,
  "holdStartDate": null,
  "appealDate": null,
  "appealReasonCode": null,
  "expirationDate": null,
  "escalationDate": null,
  "lastSellerRespDate": null,
  "lastBuyerRespDate": null,
  "refundLevelType": null,
  "refundAmount": null,
  "buyerContactedSeller": false,
  "buyerFinalAcceptRefundAmt": {
    "value": 0.0,
    "currency": "USD"
  },
  "buyerInitExpectRefundAmt": {
    "value": 0.0,
    "currency": "USD"
  },
  "buyerFinalAcceptResolution": "Unknown",
  "buyerInitExpectResolution": "BOR1 - I want a full refund (without returning
the item i.e. not received yet)",
  "creditEligible": true,
  "paypalPaid": false,
  "partialRefund": false,
  "fullRefund": false,
  "intlRefundAmount": null,
  "totalAmount": {
    "value": 101.0,
    "currency": "USD"
  },
  "refundDeadlineDate": {
    "value": "2014-09-07T21:16:50.000Z",
    "formattedValue": "2014-09-07T21:16:50.000Z"
  }
},
"inquiryHistoryDetails": {
  "history": [{
    "date": {
      "value": "2014-09-04T21:16:50.000Z",
      "formattedValue": "2014-09-04T21:16:50.000Z"
    },
    "action": "A CPS case was created."
  }
]
}

```

```

        "actor": "buyer",
        "description": "I did not receive the item.Please provide me a refund.",
        "moneyMovement": null
    }},
    "additionalInfo": "I did not receive the item.Please provide me a refund.",
    "buyerrequested": "Buyer has not received the item and wants a full refund.",
    "shipmentTrackingDetails": {
        "trackingURL": null,
        "trackingNumber": null,
        "carrier": null,
        "estimateFromDate": null,
        "estimateToDate": null,
        "currentStatus": null
    }
},
"returnDetails": null,
"extTransactionId": null
}

```

Output Container/Field	Type	Description
state	CPSStateEnum	Available values: initial pendCreated created closed
status	CPSStatusEnum	Available values: inactive open waitingBuyerResponse waitingSellerResponse closed

Output Container/Field	Type	Description
initiator, actor	PartyEnum	Available values: unknown seller buyer csr system
transactionSiteId	SiteTypeEnum	Available values: EBAY_AU Australia EBAY_CA Canada EBAY_DE Germany EBAY_MAIN main ebay US site EBAY_UK United Kingdom EBAY_US alias for EBAY_MAIN

GET /buy/order/v1/inquiry/{inquiryId}/history

Retrieve the full history of a specified inquiry for the buyer.

Note: The `GET /sell/order/v1/inquiry/{inquiryId}/history` call provides the same functionality for the seller.

Sample Request

This call has no request payload.

`https://api.ebay.com/buy/order/v1/inquiry/5002645692/history`

Sample Response

```
{
  "history": [{
    "date": {
      "value": "2014-08-29T20:18:17.000Z",
      "formattedValue": "2014-08-29T20:18:17.000Z"
    },
    "action": "A CPS case was created.",
    "actor": "buyer",
    "description": "jd hv fldm bgkl",
    "moneyMovement": null
  }, {
    "date": {
      "value": "2014-08-29T20:18:48.000Z",
      "formattedValue": "2014-08-29T20:18:48.000Z"
    },
    "action": "Auto Dispose Approve the case",
    "actor": "system",
    "description": null,
    "moneyMovement": null
  }, {
    "date": {
      "value": "2014-09-05T15:59:10.000Z",
      "formattedValue": "2014-09-05T15:59:10.000Z"
    },
    "action": "User has contacted customer support. Other party cannot see this activity.",
    "actor": "buyer",
    "description": "Please help",
    "moneyMovement": null
  }
  ],
  "additionalInfo": "jd hv fldm bgkl",
  "buyerrequested": "Buyer has not received the item and wants a full refund.",
  "shipmentTrackingDetails": {
    "trackingURL": null,
    "trackingNumber": null,
    "carrier": null,
    "estimateFromDate": null,
    "estimateToDate": null,
    "currentStatus": null
  }
}
```

Output Container/Field	Type	Description
actor	PartyEnum	Identifies which party took this action. Available values: unknown seller buyer csr system

POST /sell/order/v1/inquiry/escalation

Escalate the specified inquiry for the seller.

Note: The `POST /buy/order/v1/inquiry/escalation` call provides the same functionality for the buyer.

Sample Request

```
https://api.ebay.com/sell/order/v1/inquiry/escalation
{
  "escalationReason": "OTHERS",
  "message": "Please help",
  "inquiryId": 5002657723
}
```

Input Container/Field	Type	Occurrence	Description
escalationReason	ContactCSEnum	Required	Basis for escalation. Available values: BUYER_UNHAPPY BUYER_TROUBLE SHIPPED_ITEM OTHERS

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the inquiry has been escalated).

POST /sell/order/v1/inquiry/message

Send a message from the seller regarding the specified inquiry.

Note: The `POST /buy/order/v1/inquiry/message` call provides the same functionality for the buyer.

Sample Request

```
https://api.ebay.com/buy/order/v1/inquiry/message
{
  "inquiryId":5002661989,
  "message":"I will refund your money"
}
```

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the message has been sent).

POST /sell/order/v1/inquiry/refund

Issue a voluntary full refund from the seller to the buyer for the specified inquiry.

Sample Request

```
https://api.ebay.com/sell/order/v1/inquiry/refund
{
  "inquiryId":5002661989
}
```

Sample Response

```
{
  "refundResult": {
    "refundSource": null,
    "refundStatus": "SUCCESS"
  }
}
```

Output Container/Field	Type	Description
------------------------	------	-------------

Output Container/Field	Type	Description
refundSource	string	Available values: ECHECK INSTANT_REFUND PAYOUT
refundStatus	string	Available values: FAILED PENDING SUCCESS

POST /sell/order/v1/inquiry/refundInfo

Once the seller uploads the proof of refund, use this call to notify the buyer. The state will be updated to 170-refundInfoProvided afterwards.

Sample Request

```
https://api.ebay.com/sell/order/v1/inquiry/refundInfo
{
  "inquiryId":5002661989,
  "message":"I have uploaded proof of refund."
}
```

Sample Response

```
{
  "refundResult": {
    "refundSource": null,
    "refundStatus": "SUCCESS"
  }
}
```

POST /sell/order/v1/inquiry/shipmentInfo

Update a specified inquiry with the seller's shipping and tracking information.

Sample Request

```
https://api.ebay.com/sell/order/v1/inquiry/shipmentInfo
request body
{
```

```
"shippingCarrierName": "FedEx",
"shippingDate": "2014-09-05",
"shippingCarrierNameRecognized": true,
"shippedWithTracking": false,
"trackingNumber": "",
"proofOfShipmentUploaded": false,
"sellerComments": "testing shipping",
"inquiryId": 5002619698
}
```

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the shipping and tracking information has been submitted).

Sample Error Response

The seller provided an invalid date.

```
{
  "error": [{
    "errorId": 292025,
    "domain": "resolutionsErrorDomain",
    "severity": "ERROR",
    "category": "REQUEST",
    "message": "Shipping date has to be in the past.",
    "parameter": [{
      "value": "Mon, 12 May 2014 00:00:00 GMT",
      "name": "invalidshippingdate"
    }],
    "errorName": "InvalidShippingDate",
    "organization": "Marketplace"
  }]
}
```

GET /sell/order/v1/inquiry/{inquiryId}

Retrieve details about a specified inquiry for the seller.

Note: The `GET /buy/order/v1/inquiry/{inquiryId}` call provides the same functionality for the buyer.

Sample Request

This call has no request payload.

```
https://api.ebay.com/sell/order/v1/inquiry/5002661989
```

Sample Response

```
{
  "inquiryId": 5002661989,
  "orderId": 0,
  "itemId": 350004393501,
  "transactionId": 8378946011,
```

```

"buyer": "ebp_us_buyer_140554539108450",
"seller": "ebp_us_seller_140554537847515",
"state": "created",
"status": "open",
"claimAmount": {
  "value": 101.0,
  "currency": "USD"
},
"shippingFee": {
  "value": 0.0,
  "currency": "USD"
},
"transactionSiteId": 0,
"inquiryQuantity": 1,
"initiator": "buyer",
"smirDate": {
  "value": "2014-09-10T06:59:59.000Z",
  "formattedValue": "2014-09-10T06:59:59.000Z"
},
"viewPPTrasactionUrl":
"http://payments.qa.ebay.com/ws/eBayISAPI.dll?UnifiedCheckoutSummaryRedirect2PP&buyero
rseller=0&itemid=350004393501&transid=8378946011",
"sellerClosureReason": "undefined",
"buyerClosureReason": "undefined",
"eCheckClearanceDate": null,
"lastEscalationDateforECheck": null,
"ebpEligible": false,
"itemDetails": {
  "itemTitle": "Multi item Single seller Order PPWed Jul 16 14:16:41 MST 2014",
  "itemPrice": {
    "value": 100.0,
    "currency": "USD"
  },
  "itemPictureUrl":
"http://i.ebayimg.qa.ebay.com/00/s/MTA2NlgxNjAw/z/2oEAANKvLslTxuvE/$_1.JPG?set_id=2",
  "viewPurchasedItemUrl":
"http://cgi.qa.ebay.com/ws/eBayISAPI.dll?ViewItemVersion&item=350004393501&tid=8378946
011&view=all"
},
"inquiryDetails": {
  "creationDate": {
    "value": "2014-09-04T21:16:50.000Z",
    "formattedValue": "2014-09-04T21:16:50.000Z"
  },
  "holdEndDate": null,
  "holdStartDate": null,
  "appealDate": null,
  "appealReasonCode": null,
  "expirationDate": null,
  "escalationDate": null,
  "lastSellerRespDate": null,
  "lastBuyerRespDate": null,
  "refundLevelType": null,
  "refundAmount": null,
  "buyerContactedSeller": false,
  "buyerFinalAcceptRefundAmt": {
    "value": 0.0,
    "currency": "USD"
  },
  "buyerInitExpectRefundAmt": {
    "value": 0.0,
    "currency": "USD"
  }
}

```

```

    },
    "buyerFinalAcceptResolution": "Unknown",
    "buyerInitExpectResolution": "BOR1 - I want a full refund (without returning
the item i.e. not received yet)",
    "creditEligible": true,
    "paypalPaid": false,
    "partialRefund": false,
    "fullRefund": false,
    "intlRefundAmount": null,
    "totalAmount": {
        "value": 101.0,
        "currency": "USD"
    },
    },
    "refundDeadlineDate": {
        "value": "2014-09-07T21:19:38.000Z",
        "formattedValue": "2014-09-07T21:19:38.000Z"
    }
    },
    "inquiryHistoryDetails": {
        "history": [{
            "date": {
                "value": "2014-09-04T21:16:50.000Z",
                "formattedValue": "2014-09-04T21:16:50.000Z"
            },
            "action": "A CPS case was created.",
            "actor": "buyer",
            "description": "I did not receive the item.Please provide me a refund.",
            "moneyMovement": null
        }, {
            "date": {
                "value": "2014-09-04T21:19:38.000Z",
                "formattedValue": "2014-09-04T21:19:38.000Z"
            },
            "action": "Auto Dispose Approve the case",
            "actor": "system",
            "description": null,
            "moneyMovement": null
        }
    ],
    "additionalInfo": "I did not receive the item.Please provide me a refund.",
    "buyerrequested": "Buyer has not received the item and wants a full refund.",
    "shipmentTrackingDetails": {
        "trackingURL": null,
        "trackingNumber": null,
        "carrier": null,
        "estimateFromDate": null,
        "estimateToDate": null,
        "currentStatus": null
    }
    },
    "returnDetails": null,
    "extTransactionId": null
}

```

Output Container/Field	Type	Description
------------------------	------	-------------

Output Container/Field	Type	Description
state	CPSStateEnum	Available values: initial pendCreated created closed
status	CPSStatusEnum	Available values: inactive open waitingBuyerResponse waitingSellerResponse closed
initiator, actor	PartyEnum	Available values: unknown seller buyer csr system

Output Container/Field	Type	Description
transactionSiteId	SiteTypeEnum	Available values: EBAY_AU Australia EBAY_CA Canada EBAY_DE Germany EBAY_MAIN main ebay US site EBAY_UK United Kingdom EBAY_US alias for EBAY_MAIN

GET /sell/order/v1/inquiry/{inquiryId}/history

Retrieve the full history of a specified inquiry for the seller.

Note: The `GET /buy/order/v1/inquiry/{inquiryId}/history` call provides the same functionality for the buyer.

Sample Request

This call has no request payload.

```
https://api.ebay.com/sell/order/v1/inquiry/5002645692/history
```

Sample Response

```
{
  "history": [{
    "date": {
      "value": "2014-08-29T20:18:17.000Z",
      "formattedValue": "2014-08-29T20:18:17.000Z"
    },
    "action": "A CPS case was created.",
    "actor": "buyer",
    "description": "jdhvfldm bgkl",
    "moneyMovement": null
  }, {
    "date": {
      "value": "2014-08-29T20:18:48.000Z",
```

```

        "formattedValue": "2014-08-29T20:18:48.000Z"
    },
    "action": "Auto Dispose Approve the case",
    "actor": "system",
    "description": null,
    "moneyMovement": null
  }, {
    "date": {
      "value": "2014-09-05T15:59:10.000Z",
      "formattedValue": "2014-09-05T15:59:10.000Z"
    },
    "action": "User has contacted customer support. Other party cannot see this
activity.",
    "actor": "buyer",
    "description": "Please help",
    "moneyMovement": null
  }
},
"additionalInfo": "jdhvfldm bgkl",
"buyerrequested": "Buyer has not received the item and wants a full refund.",
"shipmentTrackingDetails": {
  "trackingURL": null,
  "trackingNumber": null,
  "carrier": null,
  "estimateFromDate": null,
  "estimateToDate": null,
  "currentStatus": null
}
}
}

```

Output Container/Field	Type	Description
actor	PartyEnum	Identifies which party took this action. Available values: unknown seller buyer csr system

Order Resolution API

POST /buy/order/v2/resolutioncasemanagement/appeal

Appeal a case to Customer Service for the buyer.

Note: The `POST /sell/order/v2/resolutioncasemanagement/appeal` call provides the same functionality for the seller.

Sample request

```
https://api.ebay.com/buy/order/v2/resolutioncasemanagement/appeal
{
  "caseId":5002664108,
  "message":"I am not happy with outcome",
  "reason":"I didn't get a refund"
}
```

Sample Response

```
{
  "appealEligibility": {
    "contactCSAvailable": true,
    "appealWindowDays": 45,
    "appealDueDate": 1413786472000,
    "appealable": true
  },
  "responseCode": "SUCCESS"
}
```

POST /buy/order/v2/resolutioncasemanagement/closure

Close a case on behalf of the buyer.

Sample Request

```
https://api.ebay.com/buy/order/v2/resolutioncasemanagement/closure
{
  "caseId":5001787081,
  "buyerCloseReason":"workedOutWithSeller",
  "buyerComments":"i would like to close"
}
```

Input Container/Field	Type	Occurrence
buyerCloseReason	BuyerCloseReason	Available values: itemArrived workedOutWithSeller wouldRatherKeepTheItem other

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 200.

Sample Error Response

The buyer did not provide a valid reason to close the case:

```
{
  errorMessage: {
    error: [
      domain: "resolutions",
      errorId: 292039,
      category: "Request",
      severity: "Error",
      message: "A valid close reason is required.",
      longMessage: "Please supply a valid close reason to process your
request.",
      actionToTake: CorrectInputData
    ]
  }
  extension: [
  ]
  ackValue : "FAILURE",
  responseStatus: 400,
  message: null
}
```

POST /buy/order/v2/resolutioncasemanagement/dispute

Escalate an INR (item not received) case or create a SNAD (significantly not as described) case on behalf of the buyer.

Note: The `POST /sell/order/v2/resolutioncasemanagement/dispute` call provides the same functionality for the seller.

Sample request

```
https://api.ebay.com/buy/order/v2/resolutioncasemanagement/dispute
{
  "originId":5002676081,
  "disputeType":"CPS_INR",
  "claimQuantity":1,
  "itemId":200004458448,
  "transactionId":8160038010
}
```

Input Container/Field	Type	Occurrence
-----------------------	------	------------

Input Container/Field	Type	Occurrence
disputeType	enum	Available values: CPS_INR CPS_SNAD
originId	string	The inquiryID for an INR case or the returnId for a SNAD case

Note: itemId and transactionId are required only for a SNAD case.

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the INR case has been escalated or the SNAD case has been created).

POST /buy/order/v2/resolutioncasemanagement/escalation

Escalate the specified case by contacting Customer Service on behalf of the buyer.

Sample Request

```
https://api.ebay.com/buy/order/v2/resolutioncasemanagement/escalation
{
  "caseId":5002675644
}
```

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the case has been escalated).

POST /buy/order/v2/resolutioncasemanagement/shipmentInfo

Update the specified case with the buyer's shipping information during the forced SNAD process.

Sample Request

```
https://api.ebay.com/buy/order/v2/resolutioncasemanagement/shipmentInfo
{
  "shippingCarrierName":"FedEx",
  "trackingNumber":12141098967,
  "caseId":5002654373
}
```

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the shipping and tracking information has been submitted).

GET /buy/order/v2/resolutioncasemanagement/{caseId}

Retrieve the full details of the specified case for the buyer.

Note: The GET /sell/order/v2/resolutioncasemanagement/{caseId} call provides the same functionality for the seller.

Sample Request

This call has no request payload.

```
https://api.ebay.com/buy/order/v2/resolutioncasemanagement/5002645692
```

Sample Response

```
{
  "caseId": 5002645692,
  "caseType": "ITEM_NOT_RECEIVED",
  "orderId": 0,
  "itemId": 200004428880,
  "transactionId": 8147050010,
  "buyer": "ebp_us_buyer_140926001708019",
  "seller": "sellerfee_seller",
  "state": "csContacted",
  "status": "buyerEscalate",
  "claimAmount": {
    "value": 81.0,
    "currency": "USD"
  },
  "shippingFee": {
    "value": 0.0,
    "currency": "USD"
  },
  "transactionSiteId": 0,
  "caseQuantity": 1,
  "initiator": "buyer",
  "creationDate": {
    "value": "2014-08-29T20:18:17.000Z",
    "formattedValue": "2014-08-29T20:18:17.000Z"
  },
  "lastModifiedDate": {
    "value": "2014-09-05T15:59:10.000Z",
    "formattedValue": "2014-09-05T15:59:10.000Z"
  },
  "extTransactionId": "6Y534764HJ266361W",
}
```

```

"viewPPTrasactionUrl":
"http://payments.qa.ebay.com/ws/eBayISAPI.dll?UnifiedCheckoutSummaryRedirect2PP&buyero
rseller=0&itemid=200004428880&transid=8147050010",
  "sellerClosureReason": "undefined",
  "buyerClosureReason": "undefined",
  "csAdjudicationBuyerOutcome": "undefined",
  "csAdjudicationSellerOutcome": "undefined",
  "escalateReason": "OTHERS",
  "appealStatus": null,
  "forceSnadEscalated": false,
  "fsnadShipToSellerByDate": null,
  "fsnadDaysToAddTrackingAfterShippingItem": 0,
  "itemDetails": {
    "itemTitle": "Single item Single seller Order PPTThu Aug 28 14:07:11 MST 2014",
    "itemPrice": {
      "value": 80.0,
      "currency": "USD"
    },
    "itemPictureUrl":
"http://i.ebayimg.qa.ebay.com/00/s/MTA2NlgxNjAw/z/MPkAANKvLslT~5oX/$_1.JPG?set_id=2",
    "viewPurchasedItemUrl":
"http://cgi.qa.ebay.com/ws/eBayISAPI.dll?ViewItemVersion&item=200004428880&tid=8147050
010&view=all"
  },
  "caseDetails": {
    "paymentAmountCurrency": null,
    "creationDate": {
      "value": "2014-08-29T20:18:17.000Z",
      "formattedValue": "2014-08-29T20:18:17.000Z"
    },
    "holdEndDate": null,
    "holdStartDate": null,
    "appealDate": null,
    "appealReasonCode": null,
    "expirationDate": null,
    "escalationDate": {
      "value": "2014-09-05T15:59:10.000Z",
      "formattedValue": "2014-09-05T15:59:10.000Z"
    },
    "lastSellerRespDate": null,
    "lastBuyerRespDate": null,
    "refundLevelType": null,
    "refundAmount": null,
    "buyerContactedSeller": false,
    "buyerFinalAcceptRefundAmt": {
      "value": 0.0,
      "currency": "USD"
    },
    "buyerInitExpectRefundAmt": {
      "value": 0.0,
      "currency": "USD"
    },
    "buyerFinalAcceptResolution": "Unknown",
    "buyerInitExpectResolution": "BOR1 - I want a full refund (without returning
the item i.e. not received yet)",
    "creditEligible": true,
    "paypalPaid": true,
    "partialRefund": false,
    "fullRefund": false,
    "refundDeadlineDate": {
      "value": "2014-09-08T15:59:10.000Z",
      "formattedValue": "2014-09-08T15:59:10.000Z"
    }
  }

```



```

    },
    "caseHistoryDetails": {
      "history": [{
        "date": {
          "value": "2014-08-29T20:18:17.000Z",
          "formattedValue": "2014-08-29T20:18:17.000Z"
        },
        "action": "A CPS case was created.",
        "actor": "buyer",
        "description": "jdhvfldm bgkl",
        "moneyMovement": null
      }, {
        "date": {
          "value": "2014-08-29T20:18:48.000Z",
          "formattedValue": "2014-08-29T20:18:48.000Z"
        },
        "action": "Auto Dispose Approve the case",
        "actor": "system",
        "description": null,
        "moneyMovement": null
      }, {
        "date": {
          "value": "2014-09-05T15:59:10.000Z",
          "formattedValue": "2014-09-05T15:59:10.000Z"
        },
        "action": "User has contacted customer support. Other party cannot see
this activity.",
        "actor": "buyer",
        "description": "Please help",
        "moneyMovement": null
      }
    ]},
    "additionalInfo": "jdhvfldm bgkl",
    "buyerrequested": "Buyer has not received the item and wants a full refund.",
    "shipmentTrackingDetails": {
      "trackingURL": null,
      "trackingNumber": null,
      "carrier": null,
      "estimateFromDate": null,
      "estimateToDate": null,
      "currentStatus": null
    }
  },
  "returnDetails": null,
  "paymentSummary": {
    "voluntaryRefundSummary": null,
    "paypalVoluntaryRefundSummary": null,
    "forceRefundSummary": null,
    "paypalForceRefundSummary": null,
    "buyerPayoutSummary": null,
    "totalBuyerRefundedAmount": null,
    "buyerRefundedAmountFromSeller": null,
    "buyerRefundedAmountFromThirdParty": null,
    "buyerRefunded": false,
    "buyerRefundedViaPaypal": false
  },
  "eligibleForAppeal": false,
  "claimDecisionId": 11007,
  "claimDecision": "EBP_C_INR_BWSF_NO_SELLER_RESPONSE",
  "minimumAmountRequireShippingConfirmation": {
    "value": 750.0,
    "currency": "USD"
  }
}

```

```

    },
    "maxDaysToFileClaim": 30,
    "numOfDaysToReturnItemForRefund": 3,
    "numDaysToExpireWithResponse": 30,
    "numDaysToExpireWithoutResponse": 30,
    "numDaysForBuyerToProvideProofOfShipment": -1,
    "contactUsLink": "http://ocs.qa.ebay.com/ws/eBayISAPI.dll?CustomerSupport",
    "buyerProtectionLink": "http://pages.qa.ebay.com/help/policies/buyer-
protection.html",
    "payForItemLink": "http://pages.qa.ebay.com/help/policies/unpaid-item.html",
    "buyerProtectedProgramLink": "http://pages.qa.ebay.com/help/buy/protection-
programs.html",
    "recalledItemLink": "http://pages.qa.ebay.com/help/policies/recalled.html",
    "priceSignatureConfirmationAmount": "$750.00",
    "numDaysForSMIR": 3,
    "eCheckClearanceDate": null,
    "eventTicketsLink": "http://pages.qa.ebay.com/help/policies/event-tickets.html"
}

```

Output Container/Field	Type	Description
claimType	CaseTypeEnum	The claim is either a regular INR, or is an "escalated return" - both part of the post transaction experience. Available values: ITEM_NOT_RECEIVED RETURN
csAdjudicationBuyerOutcome, csAdjudicationSellerOutcome	Outcome	Customer Service adjudication outcome for the buyer or the seller. Available values: win lose undefined

POST /sell/order/v2/resolutioncasemanagement/appeal

Appeal a case to Customer Service for the seller.

Note: The `POST /buy/order/v2/resolutioncasemanagement/appeal` call provides the same functionality for the buyer.

Sample request

```
https://api.ebay.com/sell/order/v2/resolutioncasemanagement/appeal
{
  "caseId":5002645692,
  "message":"i am not satisfied with outcome",
  "reason":"sellerAppealReason6"
}
```

Sample Response

```
{
  "appealEligibility": {
    "contactCSAvailable": true,
    "appealWindowDays": 45,
    "appealDueDate": 1413831165000,
    "appealable": true
  },
  "responseCode": "SUCCESS"
}
```

POST /sell/order/v2/resolutioncasemanagement/dispute

Escalate an INR (item not received) case or create a SNAD (significantly not as described) case on behalf of the seller.

Note: The `POST /buy/order/v2/resolutioncasemanagement/dispute` call provides the same functionality for the buyer.

Sample request

```
https://api.ebay.com/sell/order/v2/resolutioncasemanagement/dispute
{
  "originId":5002676123,
  "disputeType":"CPS_INR",
  "claimQuantity":1,
  "itemId":200004458456,
  "transactionId":8160045010
}
```

Input Container/Field	Type	Occurrence
disputeType	enum	Available values: CPS_INR CPS_SNAD
originId	string	The inquiryID for an INR case or the returnId for a SNAD case

Note: itemId and transactionId are required only for a SNAD case.

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201 (the INR case has been escalated or the SNAD case has been created).

POST /sell/order/v2/resolutioncasemanagement/refund

Issue a voluntary full refund from the seller to the buyer for the specified case.

Sample Request

```
https://api.ebay.com/sell/order/v2/resolutioncasemanagement/refund
{
  "caseId":5002636273,
  "message":"test refund buyer after"
}
```

Sample Response

```
{
  "refundResult": {
    "refundSource": null,
    "refundStatus": "FAILED"
  }
}
```

Output Container/Field	Type	Description
refundSource	string	Available values: ECHECK INSTANT_REFUND PAYOUT
refundStatus	string	Available values: FAILED PENDING SUCCESS

POST /sell/order/v2/resolutioncasemanagement/returnaddress

Update the specified case with the return shipping address and RMA for SNAD cases on behalf of the seller before the forced SNAD process is started.

Sample request

```
https://api.ebay.com/sell/order/v2/resolutioncasemanagement/returnaddress
{
  "firstName": "QIBUS",
  "lastName": "seller",
  "address1": "2145 Hamilton Avenue",
  "address2": "",
  "city": "San Jose",
  "state": "California",
  "zip": "95125",
  "rma": "AUTH-123",
  "countryCode": "US",
  "caseId": 5002652193
}
```

Note: The **state** is required only for countries other than DE, and if states exist for the given **countryCode**.

Sample Response

This call has no response payload. If successful, it returns an HTTP status code 201.

GET /sell/order/v2/resolutioncasemanagement/{caseId}

Retrieve the full details of the specified case for the seller.

Note: The **GET /buy/order/v2/resolutioncasemanagement/{caseId}** call provides the same functionality for the buyer.

Sample Request

This call has no request payload.

```
https://api.ebay.com/sell/order/v2/resolutioncasemanagement/5002645692
```

Sample Response

```
{
  "caseId": 5002645692,
  "caseType": "ITEM_NOT_RECEIVED",
  "orderId": 0,
  "itemId": 200004428880,
```

```

"transactionId": 8147050010,
"buyer": "ebp_us_buyer_140926001708019",
"seller": "sellerfee_seller",
"state": "csContacted",
"status": "buyerEscalate",
"claimAmount": {
  "value": 81.0,
  "currency": "USD"
},
"shippingFee": {
  "value": 0.0,
  "currency": "USD"
},
"transactionSiteId": 0,
"caseQuantity": 1,
"initiator": "buyer",
"creationDate": {
  "value": "2014-08-29T20:18:17.000Z",
  "formattedValue": "2014-08-29T20:18:17.000Z"
},
"lastModifiedDate": {
  "value": "2014-09-05T15:59:10.000Z",
  "formattedValue": "2014-09-05T15:59:10.000Z"
},
"extTransactionId": "6Y534764HJ266361W",
"viewPPTrasanctionUrl":
"http://payments.qa.ebay.com/ws/eBayISAPI.dll?UnifiedCheckoutSummaryRedirect2PP&buyero
rseller=0&itemid=200004428880&transid=8147050010",
"sellerClosureReason": "undefined",
"buyerClosureReason": "undefined",
"csAdjudicationBuyerOutcome": "undefined",
"csAdjudicationSellerOutcome": "undefined",
"escalateReason": "OTHERS",
"appealStatus": null,
"forceSnadEscalated": false,
"fsnadShipToSellerByDate": null,
"fsnadDaysToAddTrackingAfterShippingItem": 0,
"itemDetails": {
  "itemTitle": "Single item Single seller Order PPTThu Aug 28 14:07:11 MST 2014",
  "itemPrice": {
    "value": 80.0,
    "currency": "USD"
  },
  "itemPictureUrl":
"http://i.ebayimg.qa.ebay.com/00/s/MTA2NlgxNjAw/z/MPkAANKvLslT~5oX/$_1.JPG?set_id=2",
  "viewPurchasedItemUrl":
"http://cgi.qa.ebay.com/ws/eBayISAPI.dll?ViewItemVersion&item=200004428880&tid=8147050
010&view=all"
},
"caseDetails": {
  "paymentAmountCurrency": null,
  "creationDate": {
    "value": "2014-08-29T20:18:17.000Z",
    "formattedValue": "2014-08-29T20:18:17.000Z"
  },
  "holdEndDate": null,
  "holdStartDate": null,
  "appealDate": null,
  "appealReasonCode": null,
  "expirationDate": null,
  "escalationDate": {
    "value": "2014-09-05T15:59:10.000Z",

```

```

        "formattedValue": "2014-09-05T15:59:10.000Z"
    },
    "lastSellerRespDate": null,
    "lastBuyerRespDate": null,
    "refundLevelType": null,
    "refundAmount": null,
    "buyerContactedSeller": false,
    "buyerFinalAcceptRefundAmt": {
        "value": 0.0,
        "currency": "USD"
    },
    "buyerInitExpectRefundAmt": {
        "value": 0.0,
        "currency": "USD"
    },
    "buyerFinalAcceptResolution": "Unknown",
    "buyerInitExpectResolution": "BOR1 - I want a full refund (without returning
the item i.e. not received yet)",
    "creditEligible": true,
    "paypalPaid": true,
    "partialRefund": false,
    "fullRefund": false,
    "refundDeadlineDate": {
        "value": "2014-09-08T15:59:10.000Z",
        "formattedValue": "2014-09-08T15:59:10.000Z"
    }
},
"caseHistoryDetails": {
    "history": [{
        "date": {
            "value": "2014-08-29T20:18:17.000Z",
            "formattedValue": "2014-08-29T20:18:17.000Z"
        },
        "action": "A CPS case was created.",
        "actor": "buyer",
        "description": "jdhvflm bgkl",
        "moneyMovement": null
    }, {
        "date": {
            "value": "2014-08-29T20:18:48.000Z",
            "formattedValue": "2014-08-29T20:18:48.000Z"
        },
        "action": "Auto Dispose Approve the case",
        "actor": "system",
        "description": null,
        "moneyMovement": null
    }, {
        "date": {
            "value": "2014-09-05T15:59:10.000Z",
            "formattedValue": "2014-09-05T15:59:10.000Z"
        },
        "action": "User has contacted customer support. Other party cannot see
this activity.",
        "actor": "buyer",
        "description": "Please help",
        "moneyMovement": null
    }
    ],
    "additionalInfo": "jdhvflm bgkl",
    "buyerrequested": "Buyer has not received the item and wants a full refund.",
    "shipmentTrackingDetails": {
        "trackingURL": null,
        "trackingNumber": null,
    }
}

```

```

        "carrier": null,
        "estimateFromDate": null,
        "estimateToDate": null,
        "currentStatus": null
    }
},
"returnDetails": null,
"paymentSummary": {
    "voluntaryRefundSummary": null,
    "paypalVoluntaryRefundSummary": null,
    "forceRefundSummary": null,
    "paypalForceRefundSummary": null,
    "buyerPayoutSummary": null,
    "recoupmentSummary": null,
    "reimbursementSummary": null,
    "totalBuyerRefundedAmount": null,
    "buyerRefundedAmountFromSeller": null,
    "buyerRefundedAmountFromThirdParty": null,
    "buyerRefunded": false,
    "buyerRefundedViaPaypal": false
},
"eligibleForAppeal": false,
"claimDecisionId": 0,
"claimDecision": null,
"minimumAmountRequireShippingConfirmation": {
    "value": 750.0,
    "currency": "USD"
},
"maxDaysToFileClaim": 30,
"numOfDaysToReturnItemForRefund": 3,
"numDaysToExpireWithResponse": 30,
"numDaysToExpireWithoutResponse": 30,
"numDaysForBuyerToProvideProofOfShipment": -1,
"contactUsLink": "http://ocs.qa.ebay.com/ws/eBayISAPI.dll?CustomerSupport",
"buyerProtectionLink": "http://pages.qa.ebay.com/help/policies/buyer-
protection.html",
"payForItemLink": "http://pages.qa.ebay.com/help/policies/unpaid-item.html",
"buyerProtectedProgramLink": "http://pages.qa.ebay.com/help/buy/protection-
programs.html",
"recalledItemLink": "http://pages.qa.ebay.com/help/policies/recalled.html",
"priceSignatureConfirmationAmount": "$750.00",
"numDaysForSMIR": 3,
"eCheckClearanceDate": null,
"eventTicketsLink": "http://pages.qa.ebay.com/help/policies/event-tickets.html"
}

```


Output Container/Field	Type	Description
claimType	CaseTypeEnum	The claim is either a regular INR, or is an "escalated return" - both part of the post transaction experience. Available values: ITEM_NOT_RECEIVED RETURN
csAdjudicationBuyerOutcome, csAdjudicationSellerOutcome	Outcome	Customer Service adjudication outcome for the buyer or the seller. Available values: win lose undefined

Note: When **paymentSummary.reimbursementSummary** is populated, as much data as possible is being returned in the **reimbursementTransactionsList** container:

```

"paymentSummary":
{
  "reimbursementSummary":
  {
    "outstandingBalance": <Amount>,
    "reimbursementAmountPaidButNotInTransactionsList": <Amount>,
    "reimbursementTransactionsList": [
    {
      "transactionDate": <DateTime>,
      "transactionType": <String>,
      "transactionDescription": <String>,
      "amount": <Amount>,
      "contentBundleMappingId": <int>
    } ]
  },
}

```

In certain cases, **reimbursementTransactionsList** is NOT showing every reimbursement-related activity. So in such cases where the transactions don't add up to zero, and displaying such data would be confusing to the user, the service will populate the **reimbursementAmountPaidButNotInTransactionsList** field to allow consumers to display to the user all other reimbursement-related amounts. This will allow an application to display all transaction related

information such that it is not confusing to the user, who will see all the transactions, and all the amounts displayed should reconcile.

Order Return API

For a representation of the full payload structure and content of each call, see the Order Return API skeleton documentation.

GET /buy/order/v1/return/checkBillingAgreement

Check the billing agreement for a return on behalf of the buyer.

Sample request

```
https://api.ebay.com/buy/order/v1/return/checkBillingAgreement?itemId=190004238788&transactionId=8546962009&flow=MONEY_BACK
```

Query Parameter	Type	Occurrence	Available Values
flowType	ReturnType	Required	CANCEL MONEY_BACK REPLACEMENT
itemId	long	Required	
transactionId	long	Required	

Sample Response

```
{  
  "ackValue": "SUCCESS",  
  "responseStatus": 200,  
  "baOnFile": true,  
  "baSignedDate": 1406250996000,  
  "paymentBAREfId": "PGW_59252276"  
}
```

POST /buy/order/v1/return/create

Create a return on behalf of the buyer.

Sample request (without session ID)

```
https://api.ebay.com/buy/order/v1/return/create  
{
```

```

"channel": "WEB",
"creationInfo":
{
  "itemId": 190004197717,
  "transactionId": 8534772009,
  "returnQuantity": 1,
  "type": "MONEY_BACK",
  "reason": "NO_LONGER_NEED_ITEM",
  "comments": "...",
  "carrier": "USPS"
},
"includeReturnDetail": true
}

```

Sample request (with session ID)

```

https://api.ebay.com/buy/order/v1/return/create
{
  "draftId": 6213743521,
  "includeReturnDetail": true
}

```

Input Container/Field	Type	Occurrence
creationInfo	ReturnRequestType	Required
draftId	long	Required
includeReturnDetail	boolean	Optional

Sample Response

```

{
  "ackValue": "SUCCESS",
  "responseStatus": 201,
  "returnId": 5000080383,
  "detail": {
    "responseStatus": 200,
    "summary": {
      "id": 5000080383,
      "buyerLoginName": "abhi_r_buyer",
      "sellerLoginName": "abhi_r_seller",
      "currentType": "MONEY_BACK",
      "state": "RMA_PENDING",
      "status": "WAITING_FOR_RMA",
      "creationInfo": {
        "item": {
          "itemId": "190004197717",
          "transactionId": "8534772009",
          "returnQuantity": 1,
          "itemTitle": "Updated the price"
        },
        "type": "MONEY_BACK",

```

```

        "reason": "NO_LONGER_NEED_ITEM",
        "comments": "...",
        "creationDate": 1409264115000
    },
    "sellerTotalRefund": {
        "estimatedRefundAmount": {
            "value": 11.2,
            "currencyId": "USD"
        },
        "actualRefundAmount": null
    },
    "buyerTotalRefund": {
        "estimatedRefundAmount": {
            "value": 11.2,
            "currencyId": "USD"
        },
        "actualRefundAmount": null
    },
    "sellerResponseDue": {
        "activityDue": "SELLER_PROVIDE_RMA",
        "respondByDate": 1409724000000
    },
    "buyerResponseDue": {
        "activityDue": "OTHER",
        "respondByDate": 1410328800000
    },
    "escalationInfo": {
        "buyerEscalationEligibilityInfo": {
            "start": null,
            "end": null,
            "eligible": false
        },
        "sellerEscalationEligibilityInfo": {
            "start": null,
            "end": null,
            "eligible": false
        }
    },
    "sellerAvailableOption": [
        {
            "actionType": "SELLER_PROVIDE_RMA",
            "actionURL":
"http://postorder.qa.ebay.com/Return>ReturnsDetail?returnId=5000080383"
        },
        {
            "actionType": "SELLER_ISSUE_REFUND",
            "actionURL":
"http://postorder.qa.ebay.com/Return>ReturnsDetail?returnId=5000080383"
        }
    ],
    "buyerAvailableOption": [
        {
            "actionType": "BUYER_CLOSE_RETURN",
            "actionURL":
"http://postorder.qa.ebay.com/Return>ReturnsDetail?returnId=5000080383"
        }
    ],
    "returnPolicy": {
        "rmaRequired": false,
        "multiReturnAddress": true
    },
    "flags": {

```

```

        "postTransactionRtn": true,
        "escalateAvableForBuyer": true,
        "isGSP": false,
        "replacementLabelPrinted": false,
        "autoRefundTriggered": false,
        "enterpriseReturn": false,
        "requestLabel4Return": false,
        "dspReplaceSendNoShipTriggered": false,
        "canSellerDeclineReturn": false,
        "autoRouteRuleTriggered": false,
        "shippingLabelPrinted": false,
        "request4Return": false,
        "isGeo": false,
        "notifyAutoRefund": false,
        "buyerShipmentDue": false,
        "triggerByDispositionRule": false,
        "shippingLabelAvailable": true,
        "escalateAvableForSeller": false
    }
},
"detail": {
    "marketplaceId": "EBAY-US",
    "itemDetail": {
        "itemId": 190004197717,
        "transactionId": 8534772009,
        "returnQuantity": 1,
        "itemTitle": "Updated the price",
        "itemPicUrl":
"http://i.ebayimg.qa.ebay.com/00/s/MTAwN1gxNjAw/z/~9YAAN-w-0xRyKlH/$T6BCp,RshgsE-
gSbVGBLBRYKlkgKw~60_0.JPG?set_id=880000500F",
        "transactionDate": 1409262215000,
        "itemPrice": {
            "value": 14,
            "currencyId": "USD"
        }
    }
},
"buyerEmailAddress": "abhi1@unicon.ebay.com",
"sellerEmailAddress": "abhi3@unicon.ebay.com",
"checkoutType": "ONLINE",
"responseHistory": [
    {
        "author": "BUYER",
        "activity": "BUYER_CREATE_RETURN",
        "fromState": "INITIAL",
        "toState": "RMA_PENDING",
        "creationDate": 1409264115000,
        "note": "...",
        "attributes": {
            "moneyMovementRef": null,
            "trackingNumber": null,
            "carrierUsed": null,
            "partialRefundAmount": null
        }
    }
],
"refundInfo": {
    "estimatedRefundDetail": {
        "itemizedRefundDetail": [
            {
                "refundFeeType": "PURCHASE_PRICE",
                "estimatedAmount": {
                    "value": 14,

```

```

        "currencyId": "USD"
    },
    "overwritableBySeller": false,
    "amountEditable": false
},
{
    "refundFeeType": "RESTOCKING_FEE",
    "estimatedAmount": {
        "value": -2.8,
        "currencyId": "USD"
    },
    "overwritableBySeller": true,
    "amountEditable": false,
    "restockingFeePercentage": "20"
}
]
},
"actualRefundDetail": null
},
"buyerAddress": {
    "name": "QIBUS Seller",
    "street1": "2145 Hamilton Ave",
    "street2": null,
    "city": "San Jose",
    "county": null,
    "stateOrProvince": "CA",
    "country": "US",
    "postalCode": "95125-5905",
    "any": null
},
"sellerAddress": {
    "name": "test",
    "street1": "2145 Hamilton Ave",
    "street2": null,
    "city": "san jose",
    "county": null,
    "stateOrProvince": "CA",
    "country": "US",
    "postalCode": "95125-5905",
    "any": null
},
"returnShipmentInfo": {
    "shipmentTracking": {
        "shippingMethod": "SHIPPING_LABEL",
        "shippedBy": "BUYER",
        "carrierId": 2,
        "carrierEnum": "USPS",
        "carrierName": "USPS",
        "carrierUsed": "USPS",
        "deliveryStatus": "UNKNOWN",
        "toShippingAddress": {
            "name": "test",
            "street1": "2145 Hamilton Ave",
            "street2": null,
            "city": "san jose",
            "county": null,
            "stateOrProvince": "CA",
            "country": "US",
            "postalCode": "95125-5905",
            "any": null
        },
        "fromShippingAddress": {

```



```

"responseStatus": 200,
"eligibilityResult": {
  "returnItem": {
    "itemId": 220004126242,
    "transactionId": 8237293012,
    "returnQuantity": 0
  },
  "eligibilityResultsPerCheckType": [
    {
      "checkType": "RETURN_CREATION_RC",
      "eligibilityInfo": {
        "eligibilityStatus": "FAILED",
        "returnDeadline": 1395216000000,
        "eligibilityErrorDetail": [
          {
            "eligibilityError": {
              "code": "1009",
              "description": "RETURN_ALREADY_EXISTS",
              "content": "Return already exists on this transaction"
            }
          }
        ],
        "returnId": 5000075918,
        "returnCreationDate": 1393896497000
      }
    }
  ]
}
}

```

Output Container/Field	Type	Description
channelUniqueld	String	The sessionId of return draft if application
creationInfo	ReturnRequestType	Carrier is returned unless it's self-mail flow <ul style="list-style-type: none"> carrier: USPS COLLECT_PLUS ROYAL_MAIL PARCEL_FORCE reason: FOUND_BETTER_PRICE MISSING_PARTS ARRIVED_DAMAGED NO_LONGER_NEED_ITEM DEFECTIVE_ITEM EXPIRED_ITEM WRONG_SIZE requestType: REQUEST CREATE type: MONEY_BACK REPLACEMENT CANCEL
includeReturnDetail	Boolean	controls whether response will include return detail, default value is FALSE

Output Container/Field	Type	Description
requestType	Enum	Available values: REQUEST CREATE

POST /buy/order/v1/return/draft

Save draft return creation data in a database for later retrieval.

Sample request

```
https://api.ebay.com /buy/order/v1/return/draft
{
  "returnRequest": {
    "itemId": 140003980342,
    "transactionId": 8083470004,
    "returnQuantity": 1,
    "reason": "ORDERED_WRONG_ITEM",NO_LONGER_NEED_ITEM
    "type": "MONEY_BACK",
    "comments": "createDraft test"
  }
}
```

Input Container/Field	Type	Occurrence	Available Values
comments	String	Optional	
itemId	Long	Required	

Input Container/Field	Type	Occurrence	Available Values
reason	ReturnReasonType	Required	ARRIVED_DAMAGED ARRIVED_LATE BUYER_CANCEL_ORDER BUYER_NO_SHOW BUYER_NOT_SCHEDULE BUYER_REFUSED_TO_PICKUP DEFECTIVE_ITEM DIFFER_FROM_LISTING EXPIRED_ITEM FAKE_OR_COUNTERFEIT FOUND_BETTER_PRICE IN_STORE_RETURN MISSING_PARTS NO_LONGER_NEED_ITEM NO_REASON NOT_AS_DESCRIBED ORDERED_ACCIDENTALLY ORDERED_DIFFERENT_ITEM ORDERED_WRONG_ITEM OTHER OUT_OF_STOCK RETURNING_GIFT VALET_DELIVERY_ISSUES VALET_UNAVAILABLE WRONG_SIZE NO_RETURN* * This value is only applicable for the DE site

Private Release – Do Not Distribute

Input Container/Field	Type	Occurrence	Available Values
returnQuantity	Integer	Required	
transactionId	Long	Required	
type	ReturnType	Optional	CANCEL MONEY_BACK REPLACEMENT

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 201,
  "draftId": 500877800
}
```

Output Container/Field	Type	Description
ackValue	String	Indicates the status of the call.All possible values SUCCESS,PARTIAL,FAILURE
responseStatus	Integer	Commonly used status codes defined by HTTP.
draftId	Long	draft id for this new created draft record

Sample Error Response

The reason was missing from the request:

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1601,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "REQUEST",
        "message": "Missing Input",
        "parameter": [
          {
            "value": "return reason should be provided",

```

```

        "name": "parameter"
      }
    ],
    "longMessage": "Missing Input",
    "httpStatusCode": 400
  }
],
},
"ackValue": "FAILURE",
"responseStatus": 400
}

```

GET /buy/order/v1/return/draft/{draftId}

Retrieve a specified draft return for the buyer.

Sample request

```
https://api.ebay.com /buy/order/v1/return/draft/500877800
```

Input Container/Field	Type	Occurrence
draftId	long	Required

Sample Response

```

{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "returnRequest": {
    "itemId": 140003980342,
    "transactionId": 8083470004,
    "returnQuantity": 1,
    "reason": "ORDERED_WRONG_ITEM",
    "type": "MONEY_BACK",
    "comments": "createDraft test"
  }
}

```

Response Container/Field	Type	Description
comments	String	comments for return
itemId	Long	item id

Response Container/Field	Type	Description
reason	ReturnReasonType	<p>Available values:</p> <p>ARRIVED_DAMAGED</p> <p>ARRIVED_LATE</p> <p>BUYER_CANCEL_ORDER</p> <p>BUYER_NO_SHOW</p> <p>BUYER_NOT_SCHEDULE</p> <p>BUYER_REFUSED_TO_PICKUP</p> <p>DEFECTIVE_ITEM</p> <p>DIFFER_FROM_LISTING</p> <p>EXPIRED_ITEM</p> <p>FAKE_OR_COUNTERFEIT</p> <p>FOUND_BETTER_PRICE</p> <p>IN_STORE_RETURN</p> <p>MISSING_PARTS</p> <p>NO_LONGER_NEED_ITEM</p> <p>NO_REASON</p> <p>NOT_AS_DESCRIBED</p> <p>ORDERED_ACCIDENTALLY</p> <p>ORDERED_DIFFERENT_ITEM</p> <p>ORDERED_WRONG_ITEM</p> <p>OTHER</p> <p>OUT_OF_STOCK</p> <p>RETURNING_GIFT</p> <p>VALET_DELIVERY_ISSUES</p> <p>VALET_UNAVAILABLE</p> <p>WRONG_SIZE</p> <p>NO_RETURN*</p> <p>* This value is only applicable for the DE site</p>

Private Release – Do Not Distribute

Response Container/Field	Type	Description
returnQuantity	Integer	return quantity
transactionId	Long	transaction id
type	ReturnType	Available values: CANCEL MONEY_BACK REPLACEMENT

Sample Error Response

The draft ID is invalid:

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1305,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "REQUEST",
        "message": "Invalid session ID",
        "parameter": [
          {
            "value": "142",
            "name": "sessionId"
          }
        ],
        "longMessage": "Invalid session ID",
        "httpStatusCode": 400
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 400,
  "returnRequest": {
    "itemId": 0,
    "transactionId": 0,
    "returnQuantity": 0
  }
}
```

PUT /buy/order/v1/return/draft/{draftId}

Update a specified draft return on behalf of the buyer.

Sample request

```
https://api.ebay.com /buy/order/v1/return/draft/500877800
{
  "returnRequest": {
    "itemId": 140003980342,
    "reason": "ORDERED_WRONG_ITEM",
    "transactionId": 8083470004,
    "returnQuantity": 1,
    "type": "MONEY_BACK",
    "comments": "updateDraft test"
  }
}
```

Input Container/Field	Type	Occurrence	Available Values
comments	String	Optional	
draftId	long	Required	
itemId	long	Required	

Input Container/Field	Type	Occurrence	Available Values
reason	ReturnReasonType	Required	ARRIVED_DAMAGED ARRIVED_LATE BUYER_CANCEL_ORDER BUYER_NO_SHOW BUYER_NOT_SCHEDULE BUYER_REFUSED_TO_PICKUP DEFECTIVE_ITEM DIFFER_FROM_LISTING EXPIRED_ITEM FAKE_OR_COUNTERFEIT FOUND_BETTER_PRICE IN_STORE_RETURN MISSING_PARTS NO_LONGER_NEED_ITEM NO_REASON NOT_AS_DESCRIBED ORDERED_ACCIDENTALLY ORDERED_DIFFERENT_ITEM ORDERED_WRONG_ITEM OTHER OUT_OF_STOCK RETURNING_GIFT VALET_DELIVERY_ISSUES VALET_UNAVAILABLE WRONG_SIZE NO_RETURN* * This value is only applicable for the DE site

Input Container/Field	Type	Occurrence	Available Values
returnQuantity	Integer	Required	
transactionId	Long	Required	
type	ReturnType	Optional	CANCEL MONEY_BACK REPLACEMENT

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "draftId": 500877800
}
```

Output Container/Field	Type	Description
ackValue	String	Indicates the status of the call.All possible values SUCCESS,PARTIAL,FAILURE
responseStatus	Integer	Commonly used status codes defined by HTTP.
draftId	Long	draft id for this new created draft record

Sample Error Response

The reason was missing from the request:

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1601,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "REQUEST",
        "message": "Missing Input",
        "parameter": [
          {
            "value": "return reason should be provided",

```

```

        "name": "parameter"
      }
    ],
    "longMessage": "Missing Input",
    "statusCode": 400
  }
},
"ackValue": "FAILURE",
"responseStatus": 400
}

```

GET /buy/order/v1/return/eligibility

Determine if the buyer is eligible to return a specified item. Various criteria must be satisfied, including the type and timing of the return and the reason for the return.

Note: The `GET /sell/order/v1/return/eligibility` call provides the same functionality for the seller.

Sample request

```

https://api.ebay.com/buy/order/v1/return/eligibility?itemId=190004197717&checkType=ITEM_SELECTION_COMPLETE&checkType=REPLACEMENT_ITEM_SELECTION_COMPLETE&transactionId=8534772009&returnQuantity=1&reason=ORDERED_WRONG_ITEM

```

Query Parameter	Type	Occurrence	Available Values
-----------------	------	------------	------------------

Query Parameter	Type	Occurrence	Available Values
checkType	EligibilityCheckType	Required, repeatable	ITEM_SELECTION_COMPLETE ITEM_SELECTION_SIMPLE PRE_TRANSACTION REPLACEMENT_CREATION REPLACEMENT_ITEM_SELECTION_COMPLETE REPLACEMENT_ITEM_SELECTION_SIMPLE REQUEST_ITEM_SELECTION_COMPLETE REQUEST_ITEM_SELECTION_SIMPLE REQUEST_PRE_TRANSACTION REQUEST_REPLACEMENT_CREATION_COMPLETE REQUEST_REPLACEMENT_ITEM_SELECTION_COMPLETE REQUEST_RETURN_CREATION_COMPLETE REQUEST_RETURN_CREATION_SIMPLE REQUEST_UNPAID_TRANSACTION RETURN_CREATION_COMPLETE RETURN_CREATION_SIMPLE RETURN_REASON_SELECTION UNPAID_TRANSACTION
itemId	long	Required	

Query Parameter	Type	Occurrence	Available Values
reason	ReturnReasonType	Optional	ARRIVED_DAMAGED ARRIVED_LATE BUYER_CANCEL_ORDER BUYER_NO_SHOW BUYER_NOT_SCHEDULE BUYER_REFUSED_TO_PICKUP DEFECTIVE_ITEM DIFFER_FROM_LISTING EXPIRED_ITEM FAKE_OR_COUNTERFEIT FOUND_BETTER_PRICE IN_STORE_RETURN MISSING_PARTS NO_LONGER_NEED_ITEM NO_REASON NOT_AS_DESCRIBED ORDERED_ACCIDENTALLY ORDERED_DIFFERENT_ITEM ORDERED_WRONG_ITEM OTHER OUT_OF_STOCK RETURNING_GIFT VALET_DELIVERY_ISSUES VALET_UNAVAILABLE WRONG_SIZE NO_RETURN* * This value is only applicable for the DE site

Query Parameter	Type	Occurrence	Available Values
returnQuantity	Integer	Optional	
transactionId	long	Required	

Sample Response

```

{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "eligibilityResultPerItem": {
    "returnItem": {
      "itemId": 190004197717,
      "transactionId": 8534772009,
      "returnQuantity": 1
    },
    "eligibilityResultsPerCheckType": [
      {
        "checkType": "ITEM_SELECTION_COMPLETE",
        "eligibilityInfo": {
          "eligibilityStatus": "SUCCESS",
          "returnDeadline": 1411974000000,
          "sellerLoginId": "abhi_r_seller"
        }
      },
      {
        "checkType": "REPLACEMENT_ITEM_SELECTION_COMPLETE",
        "eligibilityInfo": {
          "eligibilityStatus": "SUCCESS",
          "returnDeadline": 1411974000000,
          "sellerLoginId": "abhi_r_seller"
        }
      }
    ]
  }
}

```

eligibilityStatus Enum Values

SUCCESS,
 FAILED,
 WARN

eligibilityError Enum Values

Available in 883:

Error Name	Error ID	Severity	Message
------------	----------	----------	---------

Error Name	Error ID	Severity	Message
BUYER_NOT_PAID	1001	error	"Buyer not paid"
CLASSIC_SNAD_ALREADY_EXITS	1002	error	"SNAD Case already exists on this transaction"
EBP_SNAD_ALREADY_EXITS	1003	error	"eBay Buyer Protection SNAD Case already exists on this transaction"
CLASSIC_INR_ALREADY_EXITS	1004	error	"Active INR Case exists on this transaction"
EBP_INR_ALREADY_EXITS	1005	error	"Active eBay Buyer Protection INR Case exists on this transaction"
CT_ALREADY_EXITS	1006	error	"Cancel transaction Case already exists on this transaction"
PMIS_ALREADY_EXITS	1007	error	"PMIS Case already exists on this transaction"
EBAY_CASE_ALREADY_EXITS	1008	error	"eBay Case already exists on this transaction"
RETURN_ALREADY_EXITS	1009	error	"Return already exists on this transaction"

Error Name	Error ID	Severity	Message
RETURN_NOT_ACCEPTED	1010	error	"Seller doesn't accept return policy"
RETURN_TOO_EARLY	1011	error	"Too early to file return"
RETURN_TOO_LATE	1012	error	"Past the return window"
PP_DISALLOWED_DISPUTE	1013	error	"Paypal Disallowed Dispute"
PP_ACH_RETURN	1014	error	"Paypal ACH Return"
PP_ADMINISTRATIVE_FRAUD_REVIEW	1015	error	"Administrative Fraud Review"
PP_CREDITCARD_CHARGEBACK	1016	error	"Active Paypal Credit card chargeback"
PP_ELV_REVERSAL	1017	error	"Paypal ELV Reversal"
PP_PAYPAL_GUEST	1018	error	"Paypal Guest"
PP_UNAUTHORIZED_TRANSACTION	1019	error	"Paypal unauthorized transaction"
PAYPAL_CASE_ALREADY_EXISTS	1020	error	"Active Paypal dispute"
NON_DOMESTIC_TRANSACTION	1021	error	"Non Domestic Transaction"
PAYMENT_METHOD_NOT_ACCEPTABLE	1022	error	"Payment method is not acceptable for return"

Error Name	Error ID	Severity	Message
INELIGIBLE_TRANSACTION_SITE	1023	error	"Transaction site is not eligible for filing return"
INELIGIBLE_LISTING_SITE	1024	error	"Listing site is not eligible for filing return"
EXCLUDED_CATEGORY	1025	error	"Item category is not eligible for filing return"
BEPP_EXCLUDED_CATEGORY	1026	error	"Business equipment item category is not eligible for filing return"
VPP_EXCLUDED_CATEGORY	1027	error	"Vehicle item category is not eligible for filing return"
LPV_EXISTS	1028	error	"Transaction has Local Voucher Pickup code"
FREIGHT_SHIPPING_METHOD	1029	error	"Return is not eligible as Freight shipping method is used"
NO_RESTOCKING_FEE	1030	error	"Return is not eligible as restocking fee is not selected"
SUSPENDED_SELLER	1031	error	"Return is not eligible as seller is suspended. Please file eBP SNAD case"

Error Name	Error ID	Severity	Message
ADMIN_ENDED_ITEM	1032	error	"Return is not eligible as item is admin ended. Please file eBP SNAD case"
INELIGIBLE_SELLER_REGISTRATION_SITE	1033	error	"Seller registration site is not eligible for filing return"
BUYER_REMOVED_FROM_RETURNS_PROGRAM	1034	error	"Buyer removed from the returns program"
SELLER_NOT_OPTED_RETURNS	1035	error	"Seller not opted into return policy"
INELIGIBLE_WARRANTY_ITEM	1036	error	"Return is not eligible on a warranty item"
NO_ESTIMATED_DELIVERY_DATE	1037	error	"Return is not eligible as there is no estimated delivery date"
SELLER_ISSUE_FULL_REFUND	1038	error	"Return is not eligible as enough refund amount has been issued"
SELLER_ISSUE_PARTIAL_REFUND_WITH_ENOUGH_AMT	1039	error	"Return is not eligible as enough refund amount has been issued"

Error Name	Error ID	Severity	Message
BUYER_SNAD_LIMIT_EXCEEDED	1040	error	"Return is not eligible as buyer exceeded the SNAD limit cap allowed"
REPLACEMENT_NOT_ACCEPTED	1041	error	"Seller do not accept replacement"
REFUND_AMOUNT_EXCEEDS_LIMIT	1042	error	"Refund amount exceeds the limit"
SELLER_NOT_IN_RAMP	1043	error	"Seller is not in the replacement ramp"
BOPIS Error Codes			
ONLINE_RETURN_FOR_BOPIS_CHECKOUT	1044	error	"BOPIS online return is not allowed"
STORE_CANCEL	1045	error	"Store cancelled the transaction"
INELIGIBLE_CHANNEL_FOR_ONLINE_CHECKOUT	1046	error	"Ineligible return channel for online checkout"
ONLINE_RETURN_FOR_ENOW_CHECKOUT	1047	error	"eBay Now online return is not allowed"
INVALID_REASON_FOR_REPLACEMENT	1048	error	"Invalid reason for replacement"
Application/Preliminary Check Error Codes			

Error Name	Error ID	Severity	Message
INTERNAL_SYSTEM_ERROR	2001	error	"Internal system error in processing the checks."
INVALID_TRANSACTION_ID	2002	error	"Internal system error in fetching the transaction information."
INVALID_ITEM_ID	2003	error	"Internal system error in fetching the item information."
RETURNS_PC_NOT_ENABLED	2004	error	"Returns features is not supported at this time."
VLP_NOT_SUPPORTED	2005	error	"Returns not supported for VLP items."

Available in 887:

Error Name	Error ID	Severity	Message
BUYER_NOT_PAID	1001	error	"Buyer not paid"
CLASSIC_SNAD_ALREADY_EXISTS	1002	error	"SNAD Case already exists on this transaction"
EBP_SNAD_ALREADY_EXISTS	1003	error	"eBay Buyer Protection SNAD Case already exists on this transaction"

Error Name	Error ID	Severity	Message
CLASSIC_INR_ALREADY_EXITS	1004	error	"Active INR Case exists on this transaction"
EBP_INR_ALREADY_EXITS	1005	error	"Active eBay Buyer Protection INR Case exists on this transaction"
CT_ALREADY_EXITS	1006	error	"Cancel transaction Case already exists on this transaction"
PMIS_ALREADY_EXITS	1007	error	"PMIS Case already exists on this transaction"
EBAY_CASE_ALREADY_EXITS	1008	error	"eBay Case already exists on this transaction"
RETURN_ALREADY_EXITS	1009	error	"Return already exists on this transaction"
RETURN_TOO_EARLY	1011	error	"Too early to file return"
PP_DISALLOWED_DISPUTE	1013	error	"Paypal Disallowed Dispute"
PP_ACH_RETURN	1014	error	"Paypal ACH Return"
PP_ADMINISTRATIVE_FRAUD_REVIEW	1015	error	"Administrative Fraud Review"

Error Name	Error ID	Severity	Message
PP_CREDITCARD_CHARGEBACK	1016	error	"Active Paypal Credit card chargeback"
PP_ELV_REVERSAL	1017	error	"Paypal ELV Reversal"
PP_PAYPAL_GUEST	1018	error	"Paypal Guest"
PP_UNAUTHORIZED_TRANSACTION	1019	error	"Paypal unauthorized transaction"
PAYPAL_CASE_ALREADY_EXISTS	1020	error	"Active Paypal dispute"
NON_DOMESTIC_TRANSACTION	1021	error	"Non Domestic Transaction"
PAYMENT_METHOD_NOT_ACCEPTABLE	1022	error	"Payment method is not acceptable for return"
INELIGIBLE_TRANSACTION_SITE	1023	error	"Transaction site is not eligible for filing return"
INELIGIBLE_LISTING_SITE	1024	error	"Listing site is not eligible for filing return"
EXCLUDED_CATEGORY	1025	error	"Item category is not eligible for filing return"
BEPP_EXCLUDED_CATEGORY	1026	error	"Business equipment item category is not eligible for filing return"

Error Name	Error ID	Severity	Message
VPP_EXCLUDED_CATEGORY	1027	error	"Vehicle item category is not eligible for filing return"
LPV_EXISTS	1028	error	"Transaction has Local Voucher Pickup code"
FREIGHT_SHIPPING_METHOD	1029	error	"Return is not eligible as Freight shipping method is used"
NO_RESTOCKING_FEE	1030	error	"Return is not eligible as restocking fee is not selected"
SUSPENDED_SELLER	1031	error	"Return is not eligible as seller is suspended. Please file eBP SNAD case"
ADMIN_ENDED_ITEM	1032	error	"Return is not eligible as item is admin ended. Please file eBP SNAD case"
INELIGIBLE_SELLER_REGISTRATION_SITE	1033	error	"Seller registration site is not eligible for filing return"
BUYER_REMOVED_FROM_RETURNS_PROGRAM	1034	error	"Buyer removed from the returns program"
INELIGIBLE_WARRANTY_ITEM	1036	error	"Return is not eligible on a warranty item"

Error Name	Error ID	Severity	Message
NO_ESTIMATED_DELIVERY_DATE	1037	error	"Return is not eligible as there is no estimated delivery date"
SELLER_ISSUE_FULL_REFUND	1038	error	"Return is not eligible as enough refund amount has been issued"
SELLER_ISSUE_PARTIAL_REFUND_WITH_ENOUGH_AMT	1039	error	"Return is not eligible as enough refund amount has been issued"
BUYER_SNAD_LIMIT_EXCEEDED	1040	error	"Return is not eligible as buyer exceeded the SNAD limit cap allowed"
REPLACEMENT_NOT_ACCEPTED	1041	error	"Seller do not accept replacement"
REFUND_AMOUNT_EXCEEDS_LIMIT	1042	error	"Refund amount exceeds the limit"
SELLER_NOT_IN_RAMP	1043	error	"Seller is not in the replacement ramp"
BOPIS			
ONLINE_RETURN_FOR_BOPIS_CHECKOUT	1044	error	"BOPIS online return is not allowed"

Error Name	Error ID	Severity	Message
STORE_CANCEL	1045	error	"Store cancelled the transaction"
INELIGIBLE_CHANNEL_FOR_ONLINE_CHECKOUT	1046	error	"Ineligible return channel for online checkout"
ONLINE_RETURN_FOR_ENOW_CHECKOUT	1047	error	"eBay Now online return is not allowed"
INVALID_REASON_FOR_REPLACEMENT	1048	error	"Invalid reason for replacement"
RETURN_REQUEST_BLOCKED	1049	error	"Return and request are blocked"
NON_DOMESTIC_TRANSACTION_GEO_GSP	1050	error	"Non Domestic Transaction due to GEO or GSP"
Application/Preliminary Check Error Codes			
INTERNAL_SYSTEM_ERROR	2001	error	"Internal system error in processing the checks."
INVALID_TRANSACTION_ID	2002	error	"Internal system error in fetching the transaction information."
INVALID_ITEM_ID	2003	error	"Internal system error in fetching the item information."

Error Name	Error ID	Severity	Message
RETURNS_PC_NOT_ENABLED	2004	error	"Returns features is not supported at this time."
VLP_NOT_SUPPORTED	2005	error	"Returns not supported for VLP items."

GET /buy/order/v1/return/estimate

Estimate return shipping cost and refund detail before a return is created.

Sample request

```
https://api.ebay.com/buy/order/v1/return/estimate?reason=NO_LONGER_NEED_ITEM&itemId=190004540256&transactionId=8154427009&returnQuantity=1
```

Query Parameter	Type	Occurrence	Available Values
itemId	long	Required	

Query Parameter	Type	Occurrence	Available Values
reason	ReturnReasonType	Required	ARRIVED_DAMAGED ARRIVED_LATE BUYER_CANCEL_ORDER BUYER_NO_SHOW BUYER_NOT_SCHEDULE BUYER_REFUSED_TO_PICKUP DEFECTIVE_ITEM DIFFER_FROM_LISTING EXPIRED_ITEM FAKE_OR_COUNTERFEIT FOUND_BETTER_PRICE IN_STORE_RETURN MISSING_PARTS NO_LONGER_NEED_ITEM NO_REASON NOT_AS_DESCRIBED ORDERED_ACCIDENTALLY ORDERED_DIFFERENT_ITEM ORDERED_WRONG_ITEM OTHER OUT_OF_STOCK RETURNING_GIFT VALET_DELIVERY_ISSUES VALET_UNAVAILABLE WRONG_SIZE NO_RETURN* * This value is only applicable for the DE site

Query Parameter	Type	Occurrence	Available Values
returnQuantity	Integer	Required	
transactionId	long	Required	

Sample Response (successful)

```

{
  "responseStatus": 200,
  "returnShippingCostInfo": {
    "shipmentPayer": "SELLER",
    "carriers": [
      {
        "carrier": "USPS",
        "shippingCost": {
          "totalAmount": {
            "value": 3,
            "currencyId": "USD"
          },
          "itemizedReturnShippingCost": [
            {
              "returnShippingCostType": "LABEL_COST",
              "amount": {
                "value": 3,
                "currencyId": "USD"
              }
            }
          ]
        }
      }
    ]
  },
  "refundInfo": {
    "buyerTotalAmount": {
      "value": 18,
      "currencyId": "USD"
    },
    "sellerTotalAmount": {
      "value": 18,
      "currencyId": "USD"
    },
    "estimatedRefundDetail": {
      "itemizedRefundDetail": [
        {
          "refundFeeType": "PURCHASE_PRICE",
          "estimatedAmount": {
            "value": 20,
            "currencyId": "USD"
          },
          "overwritableBySeller": false,
          "amountEditable": false
        },
        {
          "refundFeeType": "RESTOCKING_FEE",
          "estimatedAmount": {
            "value": -2,

```


Response Container/Field	Type	Description
refundFeeType	RefundFeeType	Available values: DOMESTIC_SHIPPING IMPORT_CHARGE INTERNATIONAL_SHIPPING ORDER_ADJUSTMENT ORIGINAL_SHIPPING OTHER PURCHASE_PRICE REFUND_FROM_PAYPAL RESTOCKING_FEE RETURN_SHIPPING
estimatedAmount	Amount	estimated amount for specific refund fee type
amtEditable	boolean	indicate if refund amount can be editable by seller
maxAmount	Amount	the maximum amount seller can edit
minAmount	Amount	the minimum amount seller can edit
restockingFeePercentage	String	restocking fee's percentage of refund amount

Sample Error Response

The transactionId was missing from the request:

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1601,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "REQUEST",
        "message": "Missing Input",

```

```

        "parameter": [
            {
                "value": "transactionId should be provided",
                "name": "parameter"
            }
        ],
        "longMessage": "Missing Input",
        "httpStatusCode": 400
    }
}
},
"ackValue": "FAILURE",
"responseStatus": 400
}

```

GET /buy/order/v1/return/metadata/{siteMetadataKey}

Retrieve information about the eBay site where returns are being processed. The type of information being requested is specified by one or more **siteMetaDataKey** parameters in the call path.

Sample Request

```

https://api.ebay.com/buy/order/v1/return/metadata/RETURN_REASONS,TIMEOUT_FOR_RMA

```

Path Parameter	Type	Occurrence
siteMetaDataKey	SiteMetadataKeyType	<p>Required (you can provide more than one, comma separated). Available values:</p> <p>BUY_ESCALATE_RESOLUTION_HOURS</p> <p>ELIGIBLE_ADDRESS_SITES</p> <p>LEGAL_CONTENT_EBP_POLICY_START_DATE</p> <p>MAIL_PROVIDERS</p> <p>MARK_SHIP_DATE_RANGE</p> <p>OTHER</p> <p>POSTTXN_MAIL_PROVIDERS</p> <p>PRINT_LABEL_DIMESION</p> <p>REQUEST_REASONS</p> <p>RETURN_REASONS</p> <p>RETURN_REASONS_PUDO</p>

Path Parameter	Type	Occurrence
		SHIPPING_LABEL_DIMENSION_LIMIT SHIPPING_LABEL_WEIGHT_LIMIT SUPPRESS_MULTI_ADDRESS TIMEOUT_FOR_ESCALATION TIMEOUT_FOR_REFUND TIMEOUT_FOR_RMA TIMEOUT_FOR_SHIPPING

Sample Response

```

{
  "metadata": [
    {
      "siteMetadataKey": "RETURN_REASONS",
      "dataType": "RtnReason",
      "isArray": true,
      "value": [
        "WRONG_SIZE",
        "NO_LONGER_NEED_ITEM",
        "FOUND_BETTER_PRICE",
        "DEFECTIVE_ITEM",
        "DIFFER_FROM_LISTING",
        "MISSING_PARTS",
        "ARRIVED_DAMAGED",
        "FAKE_OR_COUNTERFEIT"
      ]
    },
    {
      "siteMetadataKey": "TIMEOUT_FOR_RMA",
      "dataType": "int",
      "isArray": false,
      "value": [
        "2"
      ]
    }
  ],
  "ackValue": "SUCCESS",
  "responseStatus": 200
}

```

GET /buy/order/v1/return/{returnId}

Retrieve a summary and details of a specified return on behalf of the buyer.

Note: The GET /sell/order/v1/return/{returnId} call provides the same functionality for the seller.

Sample Request

```
https://api.ebay.com/buy/order/v1/return/5000080362
```

Path Parameter	Type	Occurrence
returnId	long	Required

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "summary": {
    "id": 5000080383,
    "buyerLoginName": "abhi_r_buyer",
    "sellerLoginName": "abhi_r_seller",
    "currentType": "MONEY_BACK",
    "state": "RMA_PENDING",
    "status": "WAITING_FOR_RMA",
    "creationInfo": {
      "item": {
        "itemId": "190004197717",
        "transactionId": "8534772009",
        "returnQuantity": 1,
        "itemTitle": "Updated the price"
      },
      "type": "MONEY_BACK",
      "reason": "NO_LONGER_NEED_ITEM",
      "comments": "...",
      "creationDate": 1409264115000
    },
    "sellerTotalRefund": {
      "estimatedRefundAmount": {
        "value": 11.2,
        "currencyId": "USD"
      },
      "actualRefundAmount": null
    },
    "buyerTotalRefund": {
      "estimatedRefundAmount": {
        "value": 11.2,
        "currencyId": "USD"
      },
      "actualRefundAmount": null
    },
    "sellerResponseDue": {
      "activityDue": "SELLER_PROVIDE_RMA",
      "respondByDate": 1409724000000
    },
    "buyerResponseDue": {
      "activityDue": "OTHER",
```



```

        "respondByDate": 1410328800000
    },
    "escalationInfo": {
        "buyerEscalationEligibilityInfo": {
            "start": null,
            "end": null,
            "eligible": false
        },
        "sellerEscalationEligibilityInfo": {
            "start": null,
            "end": null,
            "eligible": false
        }
    },
    "sellerAvailableOption": [
        {
            "actionType": "SELLER_PROVIDE_RMA",
            "actionURL":
"http://postorder.qa.ebay.com/Return>ReturnsDetail?returnId=5000080383"
        },
        {
            "actionType": "SELLER_ISSUE_REFUND",
            "actionURL":
"http://postorder.qa.ebay.com/Return>ReturnsDetail?returnId=5000080383"
        }
    ],
    "buyerAvailableOption": [
        {
            "actionType": "BUYER_CLOSE_RETURN",
            "actionURL":
"http://postorder.qa.ebay.com/Return>ReturnsDetail?returnId=5000080383"
        }
    ],
    "returnPolicy": {
        "rmaRequired": false,
        "multiReturnAddress": true
    },
    "flags": {
        "postTransactionRtn": true,
        "escalateAvableForBuyer": true,
        "isGSP": false,
        "replacementLabelPrinted": false,
        "autoRefundTriggered": false,
        "enterpriseReturn": false,
        "requestLabel4Return": false,
        "dspReplaceSendNoShipTriggered": false,
        "canSellerDeclineReturn": false,
        "autoRouteRuleTriggered": false,
        "shippingLabelPrinted": false,
        "request4Return": false,
        "isGeo": false,
        "notifyAutoRefund": false,
        "buyerShipmentDue": false,
        "triggerByDispositionRule": false,
        "shippingLabelAvailable": true,
        "escalateAvableForSeller": false
    }
},
"detail": {
    "marketplaceId": "EBAY-US",
    "itemDetail": {
        "itemId": 190004197717,

```

```

    "transactionId": 8534772009,
    "returnQuantity": 1,
    "itemTitle": "Updated the price",
    "itemPicUrl": "http://i.ebayimg.qa.ebay.com/00/s/MTAwNlgxNjAw/z/~9YAAN-w-
0xRyKLh/$T6BCp,RshgsE-gSbVGBLBRYKlkgKw~~60_0.JPG?set_id=880000500F",
    "transactionDate": 1409262215000,
    "itemPrice": {
      "value": 14,
      "currencyId": "USD"
    }
  },
  "buyerEmailAddress": "abhil@unicon.ebay.com",
  "sellerEmailAddress": "abhi3@unicon.ebay.com",
  "checkoutType": "ONLINE",
  "responseHistory": [
    {
      "author": "BUYER",
      "activity": "BUYER_CREATE_RETURN",
      "fromState": "INITIAL",
      "toState": "RMA_PENDING",
      "creationDate": 1409264115000,
      "note": "...",
      "attributes": {
        "moneyMovementRef": null,
        "trackingNumber": null,
        "carrierUsed": null,
        "partialRefundAmount": null
      }
    }
  ],
  "refundInfo": {
    "estimatedRefundDetail": {
      "itemizedRefundDetail": [
        {
          "refundFeeType": "PURCHASE_PRICE",
          "estimatedAmount": {
            "value": 14,
            "currencyId": "USD"
          },
          "overwritableBySeller": false,
          "amountEditable": false
        },
        {
          "refundFeeType": "RESTOCKING_FEE",
          "estimatedAmount": {
            "value": -2.8,
            "currencyId": "USD"
          },
          "overwritableBySeller": true,
          "amountEditable": false,
          "restockingFeePercentage": "20"
        }
      ]
    },
    "actualRefundDetail": null
  },
  "buyerAddress": {
    "name": "QIBUS Seller",
    "street1": "2145 Hamilton Ave",
    "street2": null,
    "city": "San Jose",
    "county": null,

```

```

    "stateOrProvince": "CA",
    "country": "US",
    "postalCode": "95125-5905",
    "any": null
  },
  "sellerAddress": {
    "name": "test",
    "street1": "2145 Hamilton Ave",
    "street2": null,
    "city": "san jose",
    "county": null,
    "stateOrProvince": "CA",
    "country": "US",
    "postalCode": "95125-5905",
    "any": null
  },
  "returnShipmentInfo": {
    "shipmentTracking": {
      "shippingMethod": "SHIPPING_LABEL",
      "shippedBy": "BUYER",
      "carrierId": 2,
      "carrierEnum": "USPS",
      "carrierName": "USPS",
      "carrierUsed": "USPS",
      "deliveryStatus": "UNKNOWN",
      "toShippingAddress": {
        "name": "test",
        "street1": "2145 Hamilton Ave",
        "street2": null,
        "city": "san jose",
        "county": null,
        "stateOrProvince": "CA",
        "country": "US",
        "postalCode": "95125-5905",
        "any": null
      },
      "fromShippingAddress": {
        "name": "QIBUS Seller",
        "street1": "2145 Hamilton Ave",
        "street2": null,
        "city": "San Jose",
        "county": null,
        "stateOrProvince": "CA",
        "country": "US",
        "postalCode": "95125-5905",
        "any": null
      },
      "markAsReceived": false,
      "active": true
    },
    "shippingLabelCost": {
      "totalAmount": {
        "value": 3,
        "currencyId": "USD"
      },
      "itemizedReturnShippingCost": [
        {
          "returnShippingCostType": "LABEL_COST",
          "amount": {
            "value": 3,
            "currencyId": "USD"
          }
        }
      ]
    }
  }
}

```

```

    }
  ],
  "payee": "BUYER"
},
"closeReason": "OTHER",
"closeInfo": {
  "returnCloseReason": "OTHER",
  "buyerCloseReason": "UNKNOWN",
  "buyerCloseComment": null
}
}
}

```

POST /buy/order/v1/return/{returnId}/cancel

Cancel a specified return for the buyer.

Sample Request

```

https://api.ebay.com /buy/order/v1/return/5000075510/cancel
{
  "buyerCloseReason": "BUYER_STARTED_RETURN_ACCIDENTALLY",
  "comments": "test"
}

```

Path Parameter	Type	Occurrence
returnId	long	Required

Input Container/Field	Type	Occurrence	Available Values
-----------------------	------	------------	------------------

Input Container/Field	Type	Occurrence	Available Values
buyerCloseReason	BuyerCloseReturnReasonType	Optional	UNKNOWN BUYER_RECEIVED_REFUND BUYER_RECEIVED_REPLACEMENT BUYER_STARTED_RETURN_ACCIDENTALLY BUYER_NO_LONGER_WANT_RETURN OTHER
comments	String	Optional	

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200
}
```

POST /buy/order/v1/return/{returnId}/escalate

Escalate a specified return to customer support. Before performing this operation, you must make sure this return case is in the `itemDelivered` or `refundTimeout` state.

Sample Request

```
https://api.ebay.com/buy/order/v1/return/5000080362/escalate
```

Path Parameter	Type	Occurrence
returnId	long	Required

Input Container/Field	Type	Occurrence	Available Values
comment	string	Optional	
reason	EscalateReasonType	Optional	SELLER_NO_RESPOND TROUBLE_COMMUNICATION_WITH_SELLER TROUBLE_COMMUNICATION_WITH_BUYER NO_REFUND_RECEIVED OTHER

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "claimId": 5002089126
}
```

Sample Error Response

The return case is not in the itemDelivered or refundTimeout state:

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1502,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "APPLICATION",
        "message": "Illegal State transition",
        "parameter": [
          {
            "value": "java.lang.IllegalStateException: Illegal Activity
for the state:itemReadyToShip; activityId:buyerEscalate; party:buyer;
return:5000076129",
            "name": "message"
          }
        ],
        "longMessage": "Illegal State transition",
        "httpStatusCode": 500
      }
    ]
  },
  "ackValue": "FAILURE",
}
```

```
"responseStatus": 500,
"claimId": 0
}
```

POST /buy/order/v1/return/{returnId}/initiateShippingLabel

Initiate shipping label for a return. This will invoke shipping to get the shipping label. This operation needs to be performed before printShippingLabel.

Sample Request

```
https://api.ebay.com/buy/order/v1/return/5000080362/escalate
```

Path Parameter	Type	Occurrence
returnId	long	Required

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
}
```

Output Container/Field	Type	Description
ackValue	String	Indicates the status of the call. All possible values SUCCESS, PARTIAL, FAILURE
responseStatus	Integer	Commonly used status codes defined by HTTP.
labelURL	String	Label URL if already exists

Sample Error Response

The billing agreement is not signed:

```
{
  "errorMessage": {
```

```

    "error": [
      {
        "errorId": 1505,
        "domain": "returnErrorDomain",
        "subdomain": "Returns",
        "severity": "ERROR",
        "category": "APPLICATION",
        "message": "Missing billing agreement",
        "parameter": [],
        "errorName": "MISSING_BILLING_AGREEMENT",
        "resolution": "",
        "organization": "Marketplace",
        "errorGroups": ""
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 500
}

```

POST /buy/order/v1/return/{returnId}/markAsReceived

Mark a specified replacement as received by the buyer.

Note: As of now this operation is available for a "Replacement" case only; it will fail with an error if you try it for a "Return" case. The `POST /sell/order/v1/return/{returnId}/markAsReceived` call provides the same functionality for the seller.

Sample Request

```

https://api.ebay.com /sell/order/v1/return/5000080305/markAsReceived

{
  "comments": "rma is provided"
}

```

Input Container/Field	Type	Occurrence	Description
comments	String	Optional	Comments
returnId	Long	Required	From path parameter. ReturnId or ReplacementId depending on the case

Sample Response

```

{
  "ackValue": "SUCCESS",
  "responseStatus": 200
}

```



```
}
```

Output Container/Field	Type	Description
ackValue	String	Indicates the status of the call. Available values SUCCESS, PARTIAL, FAILURE
responseStatus	Integer	Commonly used status codes defined by HTTP.

Sample Error Response

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1502,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "APPLICATION",
        "message": "Illegal State transition",
        "parameter": [
          {
            "value": "java.lang.IllegalStateException: Illegal Activity
for the state:itemShipped; activityId:sellerProvideTracking; party:seller;
return:5000075482",
            "name": "message"
          }
        ],
        "longMessage": "Illegal State transition",
        "httpStatusCode": 500
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 500
}
```

POST /buy/order/v1/return/{returnId}/markAsShipped

Mark a specified return as shipped by the buyer. This call requests the buyer to provide or verify tracking information. If the operation is successful it will update the return to “shipped” state and update the tracking information in the returns system.

Note: The `POST /sell/order/v1/return/{returnId}/markAsShipped` call provides the same functionality for the seller.

Sample Requests

```
https://api.ebay.com/buy/order/v1/return/5000080362/markAsShipped
{
  "comments":"provide tracking number with known carrier",
  "trackingNumber":"9555644875112354665",
  "carrierEnum":"ROYAL_MAIL"
}
```

```
https://api.ebay.com/buy/order/v1/return/5000080362/markAsShipped
{
  "comments":"provide tracking number with unknown carrier",
  "trackingNumber":"9555644875112354665",
  "carrierEnum":"OTHER",
  "carrierName":"Some carrier"
}
```

```
https://api.ebay.com/buy/order/v1/return/5000080362/markAsShipped
{
  "comments":"mark as shipped no tracking available",
  "carrierEnum":"OTHER",
  "carrierName":"Some carrier",
  "shippedDate":1393920000000
}
```

Input Container/Field	Type	Occurrence	Description
-----------------------	------	------------	-------------

Input Container/Field	Type	Occurrence	Description
carrierEnum	ShippingCarrierType	Conditional	<p>Carrier used for return or replacement shipment. This field is required in countries where tracking facilities are available. Available values:</p> <p>AUSTRALIA_POST</p> <p>CANADA_POST</p> <p>COLLECT_PLUS</p> <p>DEUTSCHE_POST</p> <p>DHL</p> <p>FEDEX</p> <p>HERMES</p> <p>OTHER</p> <p>PARCEL_FORCE</p> <p>ROYAL_MAIL</p> <p>UNKNOWN</p> <p>UPS</p> <p>USPS</p>
carrierName	String	Conditional	if carrierEnum is OTHER or null, carrierName becomes mandatory
comments	String	Optional	Comments
returnId	Long	Required	From path parameter. ReturnId or ReplacementId depending on the case

Input Container/Field	Type	Occurrence	Description
ShippedDate	Date	Conditional	The shipped date is required and will be considered as proof of shipment for specific countries where tracking facilities are not available. In other countries it is not mandatory.
trackingNumber	String	Conditional	Tracking number of return or replacement shipment. It's mandatory except specific countries where tracking facilities are not available.

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200
}
```

Output Container/Field	Type	Description
ackValue	String	Indicates the status of the call. Available values SUCCESS, PARTIAL, FAILURE
responseStatus	Integer	Commonly used status codes defined by HTTP.

Sample Error Response

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1617,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "REQUEST",
        "message": "return Merchandise authorization was invalid",
        "parameter": [],
        "longMessage": "RMA input contains special character which is not allowed. Please see the documentation for list of accepted characteres for RMA. ",
        "httpStatusCode": 400
      }
    ]
  }
}
```

```

    ],
    "ackValue": "FAILURE",
    "responseStatus": 400
}

```

POST /buy/order/v1/return/{returnId}/printShippingLabel

Get the shipping label for the return. The `InitiateShippingLabel` operation should be called before invoking this operation. Returns the shipping label URL if the shipping label is ready.

Sample Request

```
https://api.ebay.com/buy/order/v1/return/5000075866/printShippingLabel
```

Path Parameter	Type	Occurrence
<code>returnId</code>	long	Required

Sample Response

```

{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "labelURL":
"http://return.qa.ebay.com/ws/eBayISAPI.dll?ResolveReturn&eventName=returnLabel&return
Id=5000075866",
  "trackingNumber": "9302020128700000317641"
}

```

Output Container/Field	Type	Description
<code>ackValue</code>	String	Indicates the status of the call. Available values are SUCCESS, PARTIAL, FAILURE
<code>responseStatus</code>	Integer	Commonly used status codes defined by HTTP.
<code>labelURL</code>	String	Label URL if already exists
<code>trackingNumber</code>	String	Tracking number of the shipment

Sample Error Response

printShippingLabel was invoked before initiateShippingLabel:

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1508,
        "domain": "returnErrorDomain",
        "subdomain": "Returns",
        "severity": "ERROR",
        "category": "APPLICATION",
        "message": "Label should be initiated before printShippingLabel.",
        "parameter": [],
        "errorName": "LABEL_NOT_INITIATED",
        "resolution": "Use initiateShippingLabel operation to initiate the
label.",
        "organization": "Marketplace",
        "errorGroups": ""
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 500
}
```

GET /buy/order/v1/return/{returnId}/trackingHistory

Get tracking activity history – all scan details of a tracking number associated with a return – on behalf of the buyer.

Note: The `GET /sell/order/v1/return/{returnId}/trackingHistory` call provides the same functionality for the seller.

Sample Request

```
https://api.ebay.com/buy/order/v1/return/5000080362/trackingHistory?
trackingNumber=0988894554621323
&carrierUsed=USPS
```

Query Parameter	Type	Occurrence	Description
carrier	String	Required	Carrier
returnId	long	Required	From returnId path parameter
trackingNumber	String	Required	TrackingNumber

Sample Response

```

{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "trackingNumber": "0988894554621323",
  "carrierUsed": "USPS",
  "trackingStatus": "DELIVERED",
  "scanHistory": [
    {
      "eventStatus": "DELIVERED",
      "eventDesc": "Delivered",
      "eventTime": "2014-04-11T12:46:00.000-07:00",
      "eventCity": "SACRAMENTO",
      "eventStateOrProvince": "CA",
      "eventPostalCode": "95834",
      "eventCode": "UNKNOWN"
    },
    {
      "eventStatus": "IN_TRANSIT",
      "eventDesc": "Arrival at Pickup point",
      "eventTime": "2014-04-10T11:47:00.000-07:00",
      "eventCity": "SACRAMENTO",
      "eventStateOrProvince": "CA",
      "eventPostalCode": "95834",
      "eventCode": "UNKNOWN"
    }
  ]
}

```

Output Container/Field	Type	Description
AckValue	String	Indicates the status of the call. All possible values SUCCESS, PARTIAL, FAILURE.
carrierUsed	String	Shipping carrier used for shipment
ResponseStatus	Integer	Commonly used status codes defined by HTTP, see Status Code List for the complete list.
scanHistory	ScanDetailType	Each activity on tracking. Its an array of scan detail
scanHistory .eventCity	String	City of the scan event

Output Container/Field	Type	Description
scanHistory .eventCode	String	
scanHistory .eventDesc	String	Description by the carrier
scanHistory .eventPostalCode	String	Post code of the scan event
scanHistory .eventStateOrProvince	String	State of the scan event
scanHistory .eventStatus	String	Status of the tracking at time of this scan event
scanHistory .eventTime	Date	Time of the event
trackingNumber	String	Tracking number
trackingStatus	String	Current status of tracking

Sample Error Response

```

{
  "errorMessage": {
    "error": [
      {
        "errorId": 1502,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "APPLICATION",
        "message": "Illegal State transition",
        "parameter": [
          {
            "value": "java.lang.IllegalStateException: Illegal Activity
for the state:itemShipped; activityId:sellerProvideTracking; party:seller;
return:5000075482",
            "name": "message"
          }
        ],
        "longMessage": "Illegal State transition",
      }
    ]
  }
}

```



```
        "httpStatusCode": 500
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 500
}
```

GET /sell/order/v1/return/eligibility

Determine if a specified item is eligible for return on behalf of the seller. Various criteria must be satisfied, including the type and timing of the return and the reason for the return.

Note: The `GET /buy/order/v1/return/eligibility` call provides the same functionality for the buyer.

Sample request

```
https://api.ebay.com/sell/order/v1/return/eligibility?itemId=190004197717&checkType=ITEM_SELECTION_COMPLETE&checkType=REPLACEMENT_ITEM_SELECTION_COMPLETE&transactionId=8534772009&returnQuantity=1&reason=ORDERED_WRONG_ITEM
```

Query Parameter	Type	Occurrence	Available Values
-----------------	------	------------	------------------

Query Parameter	Type	Occurrence	Available Values
checkType	EligibilityCheckType	Required, repeatable	ITEM_SELECTION_COMPLETE ITEM_SELECTION_SIMPLE PRE_TRANSACTION REPLACEMENT_CREATION REPLACEMENT_ITEM_SELECTION_COMPLETE REPLACEMENT_ITEM_SELECTION_SIMPLE REQUEST_ITEM_SELECTION_COMPLETE REQUEST_ITEM_SELECTION_SIMPLE REQUEST_PRE_TRANSACTION REQUEST_REPLACEMENT_CREATION_COMPLETE REQUEST_REPLACEMENT_ITEM_SELECTION_COMPLETE REQUEST_RETURN_CREATION_COMPLETE REQUEST_RETURN_CREATION_SIMPLE REQUEST_UNPAID_TRANSACTION RETURN_CREATION_COMPLETE RETURN_CREATION_SIMPLE RETURN_REASON_SELECTION UNPAID_TRANSACTION
itemId	long	Required	

Query Parameter	Type	Occurrence	Available Values
reason	ReturnReasonType	Optional	ARRIVED_DAMAGED ARRIVED_LATE BUYER_CANCEL_ORDER BUYER_NO_SHOW BUYER_NOT_SCHEDULE BUYER_REFUSED_TO_PICKUP DEFECTIVE_ITEM DIFFER_FROM_LISTING EXPIRED_ITEM FAKE_OR_COUNTERFEIT FOUND_BETTER_PRICE IN_STORE_RETURN MISSING_PARTS NO_LONGER_NEED_ITEM NO_REASON NOT_AS_DESCRIBED ORDERED_ACCIDENTALLY ORDERED_DIFFERENT_ITEM ORDERED_WRONG_ITEM OTHER OUT_OF_STOCK RETURNING_GIFT VALET_DELIVERY_ISSUES VALET_UNAVAILABLE WRONG_SIZE NO_RETURN* * This value is only applicable for the DE site

Query Parameter	Type	Occurrence	Available Values
returnQuantity	Integer	Optional	
transactionId	long	Required	

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "eligibilityResultPerItem": {
    "returnItem": {
      "itemId": 190004197717,
      "transactionId": 8534772009,
      "returnQuantity": 1
    },
    "eligibilityResultsPerCheckType": [
      {
        "checkType": "ITEM_SELECTION_COMPLETE",
        "eligibilityInfo": {
          "eligibilityStatus": "SUCCESS",
          "returnDeadline": 1411974000000,
          "sellerLoginId": "abhi_r_seller"
        }
      },
      {
        "checkType": "REPLACEMENT_ITEM_SELECTION_COMPLETE",
        "eligibilityInfo": {
          "eligibilityStatus": "SUCCESS",
          "returnDeadline": 1411974000000,
          "sellerLoginId": "abhi_r_seller"
        }
      }
    ]
  }
}
```

eligibilityStatus Enum Values

SUCCESS,
 FAILED,
 WARN

eligibilityError Enum Values

Available in 883:

Error Name	Error ID	Severity	Message
------------	----------	----------	---------

Error Name	Error ID	Severity	Message
BUYER_NOT_PAID	1001	error	"Buyer not paid"
CLASSIC_SNAD_ALREADY_EXITS	1002	error	"SNAD Case already exists on this transaction"
EBP_SNAD_ALREADY_EXITS	1003	error	"eBay Buyer Protection SNAD Case already exists on this transaction"
CLASSIC_INR_ALREADY_EXITS	1004	error	"Active INR Case exists on this transaction"
EBP_INR_ALREADY_EXITS	1005	error	"Active eBay Buyer Protection INR Case exists on this transaction"
CT_ALREADY_EXITS	1006	error	"Cancel transaction Case already exists on this transaction"
PMIS_ALREADY_EXITS	1007	error	"PMIS Case already exists on this transaction"
EBAY_CASE_ALREADY_EXITS	1008	error	"eBay Case already exists on this transaction"
RETURN_ALREADY_EXITS	1009	error	"Return already exists on this transaction"

Error Name	Error ID	Severity	Message
RETURN_NOT_ACCEPTED	1010	error	"Seller doesn't accept return policy"
RETURN_TOO_EARLY	1011	error	"Too early to file return"
RETURN_TOO_LATE	1012	error	"Past the return window"
PP_DISALLOWED_DISPUTE	1013	error	"Paypal Disallowed Dispute"
PP_ACH_RETURN	1014	error	"Paypal ACH Return"
PP_ADMINISTRATIVE_FRAUD_REVIEW	1015	error	"Administrative Fraud Review"
PP_CREDITCARD_CHARGEBACK	1016	error	"Active Paypal Credit card chargeback"
PP_ELV_REVERSAL	1017	error	"Paypal ELV Reversal"
PP_PAYPAL_GUEST	1018	error	"Paypal Guest"
PP_UNAUTHORIZED_TRANSACTION	1019	error	"Paypal unauthorized transaction"
PAYPAL_CASE_ALREADY_EXISTS	1020	error	"Active Paypal dispute"
NON_DOMESTIC_TRANSACTION	1021	error	"Non Domestic Transaction"
PAYMENT_METHOD_NOT_ACCEPTABLE	1022	error	"Payment method is not acceptable for return"

Error Name	Error ID	Severity	Message
INELIGIBLE_TRANSACTION_SITE	1023	error	"Transaction site is not eligible for filing return"
INELIGIBLE_LISTING_SITE	1024	error	"Listing site is not eligible for filing return"
EXCLUDED_CATEGORY	1025	error	"Item category is not eligible for filing return"
BEPP_EXCLUDED_CATEGORY	1026	error	"Business equipment item category is not eligible for filing return"
VPP_EXCLUDED_CATEGORY	1027	error	"Vehicle item category is not eligible for filing return"
LPV_EXISTS	1028	error	"Transaction has Local Voucher Pickup code"
FREIGHT_SHIPPING_METHOD	1029	error	"Return is not eligible as Freight shipping method is used"
NO_RESTOCKING_FEE	1030	error	"Return is not eligible as restocking fee is not selected"
SUSPENDED_SELLER	1031	error	"Return is not eligible as seller is suspended. Please file eBP SNAD case"

Error Name	Error ID	Severity	Message
ADMIN_ENDED_ITEM	1032	error	"Return is not eligible as item is admin ended. Please file eBP SNAD case"
INELIGIBLE_SELLER_REGISTRATION_SITE	1033	error	"Seller registration site is not eligible for filing return"
BUYER_REMOVED_FROM_RETURNS_PROGRAM	1034	error	"Buyer removed from the returns program"
SELLER_NOT_OPTED_RETURNS	1035	error	"Seller not opted into return policy"
INELIGIBLE_WARRANTY_ITEM	1036	error	"Return is not eligible on a warranty item"
NO_ESTIMATED_DELIVERY_DATE	1037	error	"Return is not eligible as there is no estimated delivery date"
SELLER_ISSUE_FULL_REFUND	1038	error	"Return is not eligible as enough refund amount has been issued"
SELLER_ISSUE_PARTIAL_REFUND_WITH_ENOUGH_AMT	1039	error	"Return is not eligible as enough refund amount has been issued"

Error Name	Error ID	Severity	Message
BUYER_SNAD_LIMIT_EXCEEDED	1040	error	"Return is not eligible as buyer exceeded the SNAD limit cap allowed"
REPLACEMENT_NOT_ACCEPTED	1041	error	"Seller do not accept replacement"
REFUND_AMOUNT_EXCEEDS_LIMIT	1042	error	"Refund amount exceeds the limit"
SELLER_NOT_IN_RAMP	1043	error	"Seller is not in the replacement ramp"
BOPIS Error Codes			
ONLINE_RETURN_FOR_BOPIS_CHECKOUT	1044	error	"BOPIS online return is not allowed"
STORE_CANCEL	1045	error	"Store cancelled the transaction"
INELIGIBLE_CHANNEL_FOR_ONLINE_CHECKOUT	1046	error	"Ineligible return channel for online checkout"
ONLINE_RETURN_FOR_ENOW_CHECKOUT	1047	error	"eBay Now online return is not allowed"
INVALID_REASON_FOR_REPLACEMENT	1048	error	"Invalid reason for replacement"
Application/Preliminary Check Error Codes			

Error Name	Error ID	Severity	Message
INTERNAL_SYSTEM_ERROR	2001	error	"Internal system error in processing the checks."
INVALID_TRANSACTION_ID	2002	error	"Internal system error in fetching the transaction information."
INVALID_ITEM_ID	2003	error	"Internal system error in fetching the item information."
RETURNS_PC_NOT_ENABLED	2004	error	"Returns features is not supported at this time."
VLP_NOT_SUPPORTED	2005	error	"Returns not supported for VLP items."

Available in 887:

Error Name	Error ID	Severity	Message
BUYER_NOT_PAID	1001	error	"Buyer not paid"
CLASSIC_SNAD_ALREADY_EXISTS	1002	error	"SNAD Case already exists on this transaction"
EBP_SNAD_ALREADY_EXISTS	1003	error	"eBay Buyer Protection SNAD Case already exists on this transaction"

Error Name	Error ID	Severity	Message
CLASSIC_INR_ALREADY_EXITS	1004	error	"Active INR Case exists on this transaction"
EBP_INR_ALREADY_EXITS	1005	error	"Active eBay Buyer Protection INR Case exists on this transaction"
CT_ALREADY_EXITS	1006	error	"Cancel transaction Case already exists on this transaction"
PMIS_ALREADY_EXITS	1007	error	"PMIS Case already exists on this transaction"
EBAY_CASE_ALREADY_EXITS	1008	error	"eBay Case already exists on this transaction"
RETURN_ALREADY_EXITS	1009	error	"Return already exists on this transaction"
RETURN_TOO_EARLY	1011	error	"Too early to file return"
PP_DISALLOWED_DISPUTE	1013	error	"Paypal Disallowed Dispute"
PP_ACH_RETURN	1014	error	"Paypal ACH Return"
PP_ADMINISTRATIVE_FRAUD_REVIEW	1015	error	"Administrative Fraud Review"

Error Name	Error ID	Severity	Message
PP_CREDITCARD_CHARGEBACK	1016	error	"Active Paypal Credit card chargeback"
PP_ELV_REVERSAL	1017	error	"Paypal ELV Reversal"
PP_PAYPAL_GUEST	1018	error	"Paypal Guest"
PP_UNAUTHORIZED_TRANSACTION	1019	error	"Paypal unauthorized transaction"
PAYPAL_CASE_ALREADY_EXISTS	1020	error	"Active Paypal dispute"
NON_DOMESTIC_TRANSACTION	1021	error	"Non Domestic Transaction"
PAYMENT_METHOD_NOT_ACCEPTABLE	1022	error	"Payment method is not acceptable for return"
INELIGIBLE_TRANSACTION_SITE	1023	error	"Transaction site is not eligible for filing return"
INELIGIBLE_LISTING_SITE	1024	error	"Listing site is not eligible for filing return"
EXCLUDED_CATEGORY	1025	error	"Item category is not eligible for filing return"
BEPP_EXCLUDED_CATEGORY	1026	error	"Business equipment item category is not eligible for filing return"

Error Name	Error ID	Severity	Message
VPP_EXCLUDED_CATEGORY	1027	error	"Vehicle item category is not eligible for filing return"
LPV_EXISTS	1028	error	"Transaction has Local Voucher Pickup code"
FREIGHT_SHIPPING_METHOD	1029	error	"Return is not eligible as Freight shipping method is used"
NO_RESTOCKING_FEE	1030	error	"Return is not eligible as restocking fee is not selected"
SUSPENDED_SELLER	1031	error	"Return is not eligible as seller is suspended. Please file eBP SNAD case"
ADMIN_ENDED_ITEM	1032	error	"Return is not eligible as item is admin ended. Please file eBP SNAD case"
INELIGIBLE_SELLER_REGISTRATION_SITE	1033	error	"Seller registration site is not eligible for filing return"
BUYER_REMOVED_FROM_RETURNS_PROGRAM	1034	error	"Buyer removed from the returns program"
INELIGIBLE_WARRANTY_ITEM	1036	error	"Return is not eligible on a warranty item"

Error Name	Error ID	Severity	Message
NO_ESTIMATED_DELIVERY_DATE	1037	error	"Return is not eligible as there is no estimated delivery date"
SELLER_ISSUE_FULL_REFUND	1038	error	"Return is not eligible as enough refund amount has been issued"
SELLER_ISSUE_PARTIAL_REFUND_WITH_ENOUGH_AMT	1039	error	"Return is not eligible as enough refund amount has been issued"
BUYER_SNAD_LIMIT_EXCEEDED	1040	error	"Return is not eligible as buyer exceeded the SNAD limit cap allowed"
REPLACEMENT_NOT_ACCEPTED	1041	error	"Seller do not accept replacement"
REFUND_AMOUNT_EXCEEDS_LIMIT	1042	error	"Refund amount exceeds the limit"
SELLER_NOT_IN_RAMP	1043	error	"Seller is not in the replacement ramp"
BOPIS			
ONLINE_RETURN_FOR_BOPIS_CHECKOUT	1044	error	"BOPIS online return is not allowed"

Error Name	Error ID	Severity	Message
STORE_CANCEL	1045	error	"Store cancelled the transaction"
INELIGIBLE_CHANNEL_FOR_ONLINE_CHECKOUT	1046	error	"Ineligible return channel for online checkout"
ONLINE_RETURN_FOR_ENOW_CHECKOUT	1047	error	"eBay Now online return is not allowed"
INVALID_REASON_FOR_REPLACEMENT	1048	error	"Invalid reason for replacement"
RETURN_REQUEST_BLOCKED	1049	error	"Return and request are blocked"
NON_DOMESTIC_TRANSACTION_GEO_GSP	1050	error	"Non Domestic Transaction due to GEO or GSP"
Application/Preliminary Check Error Codes			
INTERNAL_SYSTEM_ERROR	2001	error	"Internal system error in processing the checks."
INVALID_TRANSACTION_ID	2002	error	"Internal system error in fetching the transaction information."
INVALID_ITEM_ID	2003	error	"Internal system error in fetching the item information."

Error Name	Error ID	Severity	Message
RETURNS_PC_NOT_ENABLED	2004	error	"Returns features is not supported at this time."
VLP_NOT_SUPPORTED	2005	error	"Returns not supported for VLP items."

GET /sell/order/v1/return/summary

Retrieve a summary of returns by a specified buyer on behalf of the seller.

Several filters are available for this call that can restrict results by date, return status, return type, item or transaction. If more than one filter type is used in the call, boolean AND logic is applied to the result set. Boolean OR logic is used for conditions within a filter.

Sample Request

```
https://api.ebay.com/ sell/order/v1/return/summary?
  buyerLoginName=returns_test_seller
  &itemId=1390798946519
  &transactionId=123433443345
  &orderId=500065487
  &returnStateFilter=ALL_OPEN
  &creationDateRangeFrom=2013-01-31
  &creationDateRangeTo=2013-02-01
  &pageNumber=10
  &entriesPerPage=1
  &sort=+ACTUAL_AMOUNT
  &reasonTypeFilter=openEBPRequest
```

Query Parameter	Type	Occurrence	Description
buyerLoginName	String	Optional	Buyer Login Name
creationDateRangeFrom	Date	Optional	Filter criteria to search the returns created from this date
creationDateRangeTo	Date	Optional	Filter criteria to search the returns created till this date

Query Parameter	Type	Occurrence	Description
entriesPerPage	Integer	Optional	Pagination input. This parameter will be changed to limit during COS 2.0 up rev
itemId	Long	Optional	Item ID
orderId	Long	Optional	Order Management order ID
pageNumber	Integer	Optional	Page number for pagination. This parameter will be changed to offset during COS 2.0 up rev
reasonTypeFilter	String	Optional	Available values: <ul style="list-style-type: none"> • openEBPRequest • allEBPRequest • all
returnStateFilter	String	Optional	Will be mapped to Enum. Available values <ul style="list-style-type: none"> • RETURN_STARTED • ITEM_SHIPPED • ITEM_DELIVERED • CLOSED • ALL_OPEN_REPLACEMENT • ALL_OPEN (default) • ALL_OPEN_RETURN

Query Parameter	Type	Occurrence	Description
sort	String	Optional	<p>Sort the result in the specified order. We only support one sort type per request. It is a combination of sort order and sort type.</p> <p>Available sort order values:</p> <ul style="list-style-type: none"> • + (ascending - default) • - (descending) <p>Available sort type values:</p> <ul style="list-style-type: none"> • FILING_DATE (default) • REASON • STATUS • ESTIMATED_AMOUNT • BUYER_LOGIN_NAME • ACTUAL_AMOUNT • REFUND_DUE_DATE • RETURN_ID <p>Example: +ESTIMATED_AMOUNT</p>
transactionId	Long	Optional	Transaction ID

Sample Response

```
{
  "members": [
    {
      "id": 5000075782,
      "buyerLoginName": "return_tej_buyer",
      "sellerLoginName": "return_tej_seller",
      "currentType": "MONEY_BACK",
      "state": "CLOSED",
      "status": "CLOSED",
      "creationInfo": {
        "item": {
          "itemId": "350004028452",
          "transactionId": "8181222011",
          "returnQuantity": 1,
          "itemTitle": "returns item 4"
        },
        "type": "MONEY_BACK",
        "reason": "NO_LONGER_NEED_ITEM",
        "creationDate": 1393275651000
      }
    }
  ],
}
```

```

    "totalRefund": {
      "estimatedRefundAmount": {
        "value": 30,
        "currencyId": "USD"
      },
      "actualRefundAmount": {
        "value": 35,
        "currencyId": "USD"
      }
    },
    "sellerResponseDue": {
      "activityDue": "OTHER",
      "respondByDate": null
    },
    "buyerResponseDue": {
      "activityDue": "SYSTEM_NOTIFIED_SHIPPED",
      "respondByDate": 1393920000000
    },
    "returnPolicy": {
      "rmaRequired": false,
      "multiReturnAddress": false
    }
  },
  ],
  "paginationOutput":{
    "pageNumber":1,
    "entriesPerPage":10,
    "totalPages":20,
    "totalEntries":15,
    "any":[]
  },
  "countSummary":[
    {
      "count":10,
      "type":"ALL_OPEN"
    }
  ],
  "ackValue": "SUCCESS",
  "responseStatus": 200
}

```

Output Container/Field	Type	Description
AckValue	String	Indicates the status of the call. All possible values SUCCESS,PARTIAL,FAILURE.
Members	list	Contains summary details of each return that satisfies the filter criteria.

Output Container/Field	Type	Description
paginationOutput	PaginationOutput	Pagination detail as per provided by the input. If no pagination details is provided in input default values will be used
PaginationOutput .entriesPerPage	Integer	Number of returns summary per page. Default value is 1
PaginationOutput .pageNumber	Integer	Present page number for the response collection type. Default value is 1
PaginationOutput .totalEntries	Integer	No of records in the collection of response.
PaginationOutput .totalPages	Integer	Total pages for the result of filter criteria specified. i.e. totalEntries/entriesPerPage
ResponseStatus	Integer	Commonly used status codes defined by HTTP, see Status Code List for the complete list.

Sample Error Response

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1616,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "REQUEST",
        "message": "Invalid Input.",
        "parameter": [],
        "longMessage": "Invalid Input.",
        "httpStatusCode": 400
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 400
}
```

GET /sell/order/v1/return/{returnId}

Retrieve a summary and details of a specified return on behalf of the seller.

Note: The `GET /buy/order/v1/return/{returnId}` call provides the same functionality for the buyer.

Sample Request

```
https://api.ebay.com/sell/order/v1/return/5000080362
```

Path Parameter	Type	Occurrence
returnId	long	Required

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "summary": {
    "id": 5000080383,
    "buyerLoginName": "abhi_r_buyer",
    "sellerLoginName": "abhi_r_seller",
    "currentType": "MONEY_BACK",
    "state": "RMA_PENDING",
    "status": "WAITING_FOR_RMA",
    "creationInfo": {
      "item": {
        "itemId": "190004197717",
        "transactionId": "8534772009",
        "returnQuantity": 1,
        "itemTitle": "Updated the price"
      },
      "type": "MONEY_BACK",
      "reason": "NO_LONGER_NEED_ITEM",
      "comments": "...",
      "creationDate": 1409264115000
    },
    "sellerTotalRefund": {
      "estimatedRefundAmount": {
        "value": 11.2,
        "currencyId": "USD"
      },
      "actualRefundAmount": null
    },
    "buyerTotalRefund": {
      "estimatedRefundAmount": {
        "value": 11.2,
        "currencyId": "USD"
      },
      "actualRefundAmount": null
    },
    "sellerResponseDue": {
```

```

        "activityDue": "SELLER_PROVIDE_RMA",
        "respondByDate": 1409724000000
    },
    "buyerResponseDue": {
        "activityDue": "OTHER",
        "respondByDate": 1410328800000
    },
    "escalationInfo": {
        "buyerEscalationEligibilityInfo": {
            "start": null,
            "end": null,
            "eligible": false
        },
        "sellerEscalationEligibilityInfo": {
            "start": null,
            "end": null,
            "eligible": false
        }
    },
    "sellerAvailableOption": [
        {
            "actionType": "SELLER_PROVIDE_RMA",
            "actionURL":
"http://postorder.qa.ebay.com/Return>ReturnsDetail?returnId=5000080383"
        },
        {
            "actionType": "SELLER_ISSUE_REFUND",
            "actionURL":
"http://postorder.qa.ebay.com/Return>ReturnsDetail?returnId=5000080383"
        }
    ],
    "buyerAvailableOption": [
        {
            "actionType": "BUYER_CLOSE_RETURN",
            "actionURL":
"http://postorder.qa.ebay.com/Return>ReturnsDetail?returnId=5000080383"
        }
    ],
    "returnPolicy": {
        "rmaRequired": false,
        "multiReturnAddress": true
    },
    "flags": {
        "postTransactionRtn": true,
        "escalateAvableForBuyer": true,
        "isGSP": false,
        "replacementLabelPrinted": false,
        "autoRefundTriggered": false,
        "enterpriseReturn": false,
        "requestLabel4Return": false,
        "dspReplaceSendNoShipTriggered": false,
        "canSellerDeclineReturn": false,
        "autoRouteRuleTriggered": false,
        "shippingLabelPrinted": false,
        "request4Return": false,
        "isGeo": false,
        "notifyAutoRefund": false,
        "buyerShipmentDue": false,
        "triggerByDispositionRule": false,
        "shippingLabelAvailable": true,
        "escalateAvableForSeller": false
    }
}

```

```

    },
    "detail": {
      "marketplaceId": "EBAY-US",
      "itemDetail": {
        "itemId": 190004197717,
        "transactionId": 8534772009,
        "returnQuantity": 1,
        "itemTitle": "Updated the price",
        "itemPicUrl": "http://i.ebayimg.qa.ebay.com/00/s/MTAwNlgxNjAw/z/~9YAAN-w-0xRyKLh/$T6BCp,RshgsE-gSbVGbLBRyKLkwgKw~60_0.JPG?set_id=880000500F",
        "transactionDate": 1409262215000,
        "itemPrice": {
          "value": 14,
          "currencyId": "USD"
        }
      }
    },
    "buyerEmailAddress": "abhi1@unicon.ebay.com",
    "sellerEmailAddress": "abhi3@unicon.ebay.com",
    "checkoutType": "ONLINE",
    "responseHistory": [
      {
        "author": "BUYER",
        "activity": "BUYER_CREATE_RETURN",
        "fromState": "INITIAL",
        "toState": "RMA_PENDING",
        "creationDate": 1409264115000,
        "note": "...",
        "attributes": {
          "moneyMovementRef": null,
          "trackingNumber": null,
          "carrierUsed": null,
          "partialRefundAmount": null
        }
      }
    ],
    "refundInfo": {
      "estimatedRefundDetail": {
        "itemizedRefundDetail": [
          {
            "refundFeeType": "PURCHASE_PRICE",
            "estimatedAmount": {
              "value": 14,
              "currencyId": "USD"
            },
            "overwritableBySeller": false,
            "amountEditable": false
          },
          {
            "refundFeeType": "RESTOCKING_FEE",
            "estimatedAmount": {
              "value": -2.8,
              "currencyId": "USD"
            },
            "overwritableBySeller": true,
            "amountEditable": false,
            "restockingFeePercentage": "20"
          }
        ]
      },
      "actualRefundDetail": null
    },
    "buyerAddress": {

```

```

    "name": "QIBUS Seller",
    "street1": "2145 Hamilton Ave",
    "street2": null,
    "city": "San Jose",
    "county": null,
    "stateOrProvince": "CA",
    "country": "US",
    "postalCode": "95125-5905",
    "any": null
  },
  "sellerAddress": {
    "name": "test",
    "street1": "2145 Hamilton Ave",
    "street2": null,
    "city": "san jose",
    "county": null,
    "stateOrProvince": "CA",
    "country": "US",
    "postalCode": "95125-5905",
    "any": null
  },
  "returnShipmentInfo": {
    "shipmentTracking": {
      "shippingMethod": "SHIPPING_LABEL",
      "shippedBy": "BUYER",
      "carrierId": 2,
      "carrierEnum": "USPS",
      "carrierName": "USPS",
      "carrierUsed": "USPS",
      "deliveryStatus": "UNKNOWN",
      "toShippingAddress": {
        "name": "test",
        "street1": "2145 Hamilton Ave",
        "street2": null,
        "city": "san jose",
        "county": null,
        "stateOrProvince": "CA",
        "country": "US",
        "postalCode": "95125-5905",
        "any": null
      },
      "fromShippingAddress": {
        "name": "QIBUS Seller",
        "street1": "2145 Hamilton Ave",
        "street2": null,
        "city": "San Jose",
        "county": null,
        "stateOrProvince": "CA",
        "country": "US",
        "postalCode": "95125-5905",
        "any": null
      },
      "markAsReceived": false,
      "active": true
    },
    "shippingLabelCost": {
      "totalAmount": {
        "value": 3,
        "currencyId": "USD"
      },
      "itemizedReturnShippingCost": [
        {

```



```

        "returnShippingCostType": "LABEL_COST",
        "amount": {
            "value": 3,
            "currencyId": "USD"
        }
    }
},
"payee": "BUYER"
},
"closeReason": "OTHER",
"closeInfo": {
    "returnCloseReason": "OTHER",
    "buyerCloseReason": "UNKNOWN",
    "buyerCloseComment": null
}
}
}

```

POST /sell/order/v1/return/{returnId}/authorize

This operation enables sellers to authorize a return by submitting the necessary information from seller preferences. Currently it supports a return merchandise authorization when the seller indicates an RMA is required and provides a return address in the multiple address option in seller preferences.

In the future, for any action required by the seller to proceed with return will be part of this operation.

An RMA authorization code can contain only [0-9], [a-z], [A-Z], +, -, .(dot), (space), *, /, \$, % .

Other characters are not allowed and will throw an error if the input contains one.

Sample request

```

https://api.ebay.com /sell/order/v1/return/5000080362/authorize
{
  "returnMerchandiseAuthorization": "RMA_123",
  "comments": "rma is provided",
  "sellerReturnAddress": {
    "name": "Jack Sparrow",
    "street1": "2065 Hamilton Avenue",
    "street2": "Shipping Service",
    "city": "San Jose",
    "county": "Santa Clara County",
    "stateOrProvince": "California",
    "country": "United States",
    "postalCode": "95134",
    "any": []
  },
  "decision": "APPROVE",
  "partialRefundAmount": {
    "value": 102.00,
    "currencyId": "USD"
  },
}

```

Input Container/Field	Type	Occurrence	Description
comments	String	Optional	Comments
decision	DecisionType	Required	Available values: ACCEPT_PARTIAL_REFUND APPROVE APPROVE_WITHOUT_EBAY_LABEL DECLINE DECLINE_PARTIAL_REFUND OFFER_PARTIAL_REFUND OFFER_RETURN OFFER_RETURN_WITHOUT_LABEL PROVIDE_RMA UNKNOWN
partialRefundAmount	Amount	Conditional	Partial refund amount which seller would like to offer. Mandatory if a decision type of OFFER_PARTIAL_REFUND is selected.
returnId	Long	Required	From path parameter. ReturnId or ReplacementId depending on the case
returnMerchandiseAuthorization	String	Optional	Return Merchandise Authorization information
sellerReturnAddress	Address	Optional	Address data type.
comments	String	Optional	Comments

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "refundStatus": "SUCCESS"
}
```

Output Container/Field	Type	Description
ackValue	String	Indicates the status of the call. Available values SUCCESS, PARTIAL, FAILURE
responseStatus	Integer	Commonly used status codes defined by HTTP.
refundStatus	Enum	In case of buyer accepting partial refund. This parameter will determine what is the status of actual refund/money movement. Available values: REQUESTED, SUCCESS, PENDING, FAILED, UNKNOWN

Sample Error Response

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1617,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "REQUEST",
        "message": "return Merchandise authorization was invalid",
        "parameter": [],
        "longMessage": "RMA input contains special character which is not allowed. Please see the documentation for list of accepted characteres for RMA. ",
        "httpStatusCode": 400
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 400
}
```

POST /sell/order/v1/return/{returnId}/escalate

Escalate to customer support. Before performing this operation, you must make sure this return case is in the `itemDelivered` or `refundTimeout` state.

Sample Request

```
https://api.ebay.com/sell/order/v1/return/5000080362/escalate
```

Query Path Parameter	Type	Occurrence
<code>returnId</code>	long	Required

Input Container/Field	Type	Occurrence	Available Values
<code>comment</code>	string	Optional	
<code>reason</code>	EscalateReasonType	Optional	SELLER_NO_RESPOND TROUBLE_COMMUNICATION_WITH_SELLER TROUBLE_COMMUNICATION_WITH_BUYER NO_REFUND_RECEIVED OTHER

Sample Response

```
{  
  "ackValue": "SUCCESS",  
  "responseStatus": 200,  
  "claimId": 5002089126  
}
```

Output Container/Field	Type	Description
<code>ackValue</code>	String	Indicates the status of the call. Available values: SUCCESS,

Output Container/Field	Type	Description
		PARTIAL, FAILURE .
responseStatus	Integer	Commonly used HTTP status codes.
claimId	long	The identifier for this escalation case.

Sample Error Response

The return case is not in the `itemDelivered` or `refundTimeout` state:

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1502,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "APPLICATION",
        "message": "Illegal State transition",
        "parameter": [
          {
            "value": "java.lang.IllegalStateException: Illegal Activity
for the state:itemReadyToShip; activityId:buyerEscalate; party:buyer;
return:5000076129",
            "name": "message"
          }
        ],
        "longMessage": "Illegal State transition",
        "httpStatusCode": 500
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 500,
  "claimId": 0
}
```

POST /sell/order/v1/return/{returnId}/issueRefund

This call issues a refund to the buyer for a return.

Sample Request

```
https://api.ebay.com/sell/order/v1/return/5000080362/issueRefund

{
  "comments": "Refund",
}
```

```

"refundDetail":{
  "totalAmount": {
    "value":102.00,
    "currencyId":"USD"
  },
  "itemizedRefundDetail":[
    {
      "refundFeeType":"PURCHASE_PRICE",
      "amount": {
        "value":100.00,
        "currencyId":"USD"
      },
      "restockingFeePercentage":"0"
    },
    {
      "refundFeeType":"ORIGINAL_SHIPPING",
      "amount": {
        "value":2.00,
        "currencyId":"USD"
      },
      "restockingFeePercentage":"0"
    }
  ]
}

```

Input Container/Field	Type	Occurrence	Description
comments	String	Optional	Comments
itemizedRefundDetail	<u>ItemizedRefundDetailType</u>	Required, repeatable	List of itemized refund detail with the type of fee refunded, amount and restocking fee for each of them
itemizedRefundDetail .amount	Amount	Required	It consist of value and currencyId same as provided in totalAmount.

Input Container/Field	Type	Occurrence	Description
itemizedRefundDetail.refundFeeType	String	Required	Enum for refund fee type. Available values: DOMESTIC_SHIPPING IMPORT_CHARGE INTERNATIONAL_SHIPPING ORDER_ADJUSTMENT ORIGINAL_SHIPPING OTHER PURCHASE_PRICE REFUND_FROM_PAYPAL RESTOCKING_FEE RETURN_SHIPPING
itemizedRefundDetail.restockingFeePercentage	String	Conditional	Percentage of amount will be deducted from refund in case where restocking fee is applied.
returnId	Long	Required	From path parameter. ReturnId or ReplacementId depending on the case
totalAmount.currencyId	String	Required	Enum value for the currencyId. Please refer eBay generic enums for complete list of possible values
totalAmount.value	Double	Required	Total refund amount. Needs to be double

Sample Response

```
{
  "refundStatus": "SUCCESS",
  "ackValue": "SUCCESS",
  "responseStatus": 200
}
```

```
}
```

Output Container/Field	Type	Description
ackValue	String	Indicates the status of the call. Available values: SUCCESS, PARTIAL, FAILURE
refundStatus	String	Indicates the status of the refund. Available values: SUCCESS, FAILED, PENDING
responseStatus	Integer	Commonly used status codes defined by HTTP.

Sample Error Response

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1001,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "APPLICATION",
        "message": "Application encountered an internal error.",
        "parameter": [
          {
            "value": ";Duplicate warning.",
            "name": "message"
          }
        ],
        "longMessage": "Application encountered an internal error.",
        "httpStatusCode": 500
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 500,
  "refundStatus": "FAILED"
}
```

POST /sell/order/v1/return/{returnId}/markAsReceived

Mark a specified return as received by the seller.

Note: As of now this operation is available for a "Replacement" case only; it will fail with an error if you try it for a "Return" case. The `POST /buy/order/v1/return/{returnId}/markAsReceived` call provides the same functionality for the buyer.

Sample Request

```
https://api.ebay.com/sell/order/v1/return/5000080305/markAsReceived

{
  "comments": "rma is provided"
}
```

Input Container/Field	Type	Occurrence	Description
comments	String	Optional	Comments
returnId	Long	Required	From path parameter. ReturnId or ReplacementId depending on the case

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200
}
```

Output Container/Field	Type	Description
ackValue	String	Indicates the status of the call. Available values SUCCESS, PARTIAL, FAILURE
responseStatus	Integer	Commonly used status codes defined by HTTP.

Sample Error Response

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1502,
        "domain": "returnErrorDomain",
        "severity": "ERROR",

```

```

        "category": "APPLICATION",
        "message": "Illegal State transition",
        "parameter": [
            {
                "value": "java.lang.IllegalStateException: Illegal Activity
for the state:itemShipped; activityId:sellerProvideTracking; party:seller;
return:5000075482",
                "name": "message"
            }
        ],
        "longMessage": "Illegal State transition",
        "httpStatusCode": 500
    }
}
},
"ackValue": "FAILURE",
"responseStatus": 500
}

```

POST /sell/order/v1/return/{returnId}/markAsShipped

Mark a specified replacement as shipped by the seller. This call requests the seller to provide or verify tracking information. If the operation is successful it will update the replacement to “shipped” state and update the tracking information in the returns system.

Note: The `POST /buy/order/v1/return/{returnId}/markAsShipped` call provides the same functionality for the buyer.

Sample Requests

```
https://api.ebay.com/sell/order/v1/return/5000080362/markAsShipped
```

```

{
  "comments": "provide tracking number with known carrier",
  "trackingNumber": "9555644875112354665",
  "carrierEnum": "ROYAL_MAIL"
}

```

```
https://api.ebay.com/sell/order/v1/return/5000080362/markAsShipped
```

```

{
  "comments": "provide tracking number with unknown carrier",
  "trackingNumber": "9555644875112354665",
  "carrierEnum": "OTHER",
  "carrierName": "Some carrier"
}

```

```
https://api.ebay.com/sell/order/v1/return/5000080362/markAsShipped
```

```

{
  "comments": "mark as shipped no tracking available",
  "carrierEnum": "OTHER",
  "carrierName": "Some carrier",
}

```

```
"shippedDate":1393920000000
}
```

Input Container/Field	Type	Occurrence	Description
carrierEnum	ShippingCarrierType	Conditional	Carrier used for return or replacement shipment. This field is required in countries where tracking facilities are available. Available values: AUSTRALIA_POST CANADA_POST COLLECT_PLUS DEUTSCHE_POST DHL FEDEX HERMES OTHER PARCEL_FORCE ROYAL_MAIL UNKNOWN UPS USPS
carrierName	String	Conditional	if carrierEnum is OTHER or null, carrierName becomes mandatory
comments	String	Optional	Comments
returnId	Long	Required	From path parameter. ReturnId or ReplacementId depending on the case

Input Container/Field	Type	Occurrence	Description
ShippedDate	Date	Conditional	The shipped date is required and will be considered as proof of shipment for specific countries where tracking facilities are not available. In other countries it is not mandatory.
trackingNumber	String	Conditional	Tracking number of return or replacement shipment. It's mandatory except specific countries where tracking facilities are not available.

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200
}
```

Output Container/Field	Type	Description
ackValue	String	Indicates the status of the call. Available values SUCCESS, PARTIAL, FAILURE
responseStatus	Integer	Commonly used status codes defined by HTTP.

Sample Error Response

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1617,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "REQUEST",
        "message": "return Merchandise authorization was invalid",
        "parameter": [],
        "longMessage": "RMA input contains special character which is not allowed. Please see the documentation for list of accepted characteres for RMA. ",
        "httpStatusCode": 400
      }
    ]
  }
}
```

```

    ],
    "ackValue": "FAILURE",
    "responseStatus": 400
}

```

GET /sell/order/v1/return/{returnId}/trackingHistory

Get tracking activity history – all scan details of a tracking number associated with a return – on behalf of the seller.

Note: The `GET /buy/order/v1/return/{returnId}/trackingHistory` call provides the same functionality for the buyer.

Sample Request

```

https://api.ebay.com/sell/order/v1/return/5000080362/trackingHistory?
trackingNumber=0988894554621323
&carrierUsed=USPS

```

Query Parameter	Type	Occurrence	Description
carrier	String	Required	Carrier
returnId	long	Required	From returnId path parameter
trackingNumber	String	Required	TrackingNumber

Sample Response

```

{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
  "trackingNumber": "0988894554621323",
  "carrierUsed": "USPS",
  "trackingStatus": "DELIVERED",
  "scanHistory": [
    {
      "eventStatus": "DELIVERED",
      "eventDesc": "Delivered",
      "eventTime": "2014-04-11T12:46:00.000-07:00",
      "eventCity": "SACRAMENTO",
      "eventStateOrProvince": "CA",
      "eventPostalCode": "95834",
      "eventCode": "UNKNOWN"
    },
    {
      "eventStatus": "IN_TRANSIT",
      "eventDesc": "Arrival at Pickup point",

```

```

    "eventTime": "2014-04-10T11:47:00.000-07:00",
    "eventCity": "SACRAMENTO",
    "eventStateOrProvince": "CA",
    "eventPostalCode": "95834",
    "eventCode": "UNKNOWN"
  }
}

```

Output Container/Field	Type	Description
AckValue	String	Indicates the status of the call. All possible values SUCCESS, PARTIAL, FAILURE.
carrierUsed	String	Shipping carrier used for shipment
ResponseStatus	Integer	Commonly used status codes defined by HTTP, see Status Code List for the complete list.
scanHistory	ScanDetailType	Each activity on tracking. Its an array of scan detail
scanHistory .eventCity	String	City of the scan event
scanHistory .eventCode	String	
scanHistory .eventDesc	String	Description by the carrier
scanHistory .eventPostalCode	String	Post code of the scan event
scanHistory .eventStateOrProvince	String	State of the scan event
scanHistory .eventStatus	String	Status of the tracking at time of this scan event

Output Container/Field	Type	Description
scanHistory .eventTime	Date	Time of the event
trackingNumber	String	Tracking number
trackingStatus	String	Current status of tracking

Sample Error Response

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1502,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "APPLICATION",
        "message": "Illegal State transition",
        "parameter": [
          {
            "value": "java.lang.IllegalStateException: Illegal Activity
for the state:itemShipped; activityId:sellerProvideTracking; party:seller;
return:5000075482",
            "name": "message"
          }
        ],
        "longMessage": "Illegal State transition",
        "httpStatusCode": 500
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 500
}
```

POST /sell/order/v1/return/{returnId}/markRefundSent

This call is used by the seller to indicate that the buyer's refund for the returned item has been sent.

Sample Request

```
https://api.ebay.com/sell/order/v1/return/5000080362/markRefundSent
```

Query Parameter	Type	Occurrence	Description
-----------------	------	------------	-------------

Query Parameter	Type	Occurrence	Description
returnId	long	Required	From returnId path parameter

Sample Request

```

/sell/order/v1/return/5000080362/markRefundSent

{
  "comments": "Refund",
  "refundDetail": {
    "totalAmount": {
      "value":102.00,
      "currencyId":"USD"
    },
    "itemizedRefundDetail":[
      {
        "refundFeeType":"PURCHASE_PRICE",
        "amount": {
          "value":100.00,
          "currencyId":"USD"
        }
        "restockingFeePercentage":"0"
      },
      {
        "refundFeeType":"ORIGINAL_SHIPPING",
        "amount": {
          "value":2.00,
          "currencyId":"USD"
        }
        "restockingFeePercentage":"0"
      }
    ]
  }
}

```

Input Container/Field	Type	Occurrence	Description
comments	String	Optional	Comments
totalAmount.value	Double	Required	Total refund amount, including purchase price and original shipping
totalAmount.currencyId	String	Required	eBay enumeration value to denote the type of currency used

Input Container/Field	Type	Occurrence	Description
itemizedRefundDetail[]	ItemizeRefundDetailType	Required {1..*}	List of separate types of cost refunded, such as purchase price, original shipping cost, and any restocking fee
itemizedRefundDetail.refundFeeType	String	Required	Enumeration value denoting the refund fee type. Supported values for this call include: PURCHASE_PRICE, ORIGINAL_SHIPPING, IMPORT_CHARGE, RESTOCKING_FEE, RETURN_SHIPPING, ORDER_ADJUSTMENT, OTHER
ItemizedRefundDetail.amount	Amount	Required	Container consisting of the dollar value and the currency used for the refund fee type
ItemizedRefundDetail.restockingFeePercentage	String	Optional	The percentage of the purchase price that will be deducted from the seller's refund based on a restocking fee that was set for the item

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
}
```

Output Container/Field	Type	Description
------------------------	------	-------------

Output Container/Field	Type	Description
AckValue	String	Indicates the status of the call.All possible values SUCCESS,PARTIAL,FAILURE.
ResponseStatus	Integer	Commonly used status codes defined by HTTP, see Status Code List for the complete list.

Sample Error Response

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1001,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "APPLICATION",
        "message": "Application encountered an internal error.",
        "parameter": [
          {
            "value": "Duplicate warning",
            "name": "message"
          }
        ],
        "longMessage": " Application encountered an internal error.",
        "httpStatusCode": 500
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 500
}
```

POST /buy/order/v1/return/{returnId}/markRefundReceived

This call is used to mark the refund for the return as received by the buyer.

Sample Request

```
https://api.ebay.com/buy/order/v1/return/5000080362/markRefundReceived
```

Query Parameter	Type	Occurrence	Description
returnId	long	Required	From returnId path parameter

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
}
```

Output Container/Field	Type	Description
AckValue	String	Indicates the status of the call.All possible values SUCCESS,PARTIAL,FAILURE.
ResponseStatus	Integer	Commonly used status codes defined by HTTP, see Status Code List for the complete list.

Sample Error Response

```
{
  "errorMessage": {
    "error": [
      {
        "errorId": 1001,
        "domain": "returnErrorDomain",
        "severity": "ERROR",
        "category": "APPLICATION",
        "message": "Application encountered an internal error.",
        "parameter": [
          {
            "value": "Duplicate warning",
            "name": "message"
          }
        ],
        "longMessage": " Application encountered an internal error.",
        "httpStatusCode": 500
      }
    ]
  },
  "ackValue": "FAILURE",
  "responseStatus": 500
}
```

POST /buy/order/v1/return/{returnId}/sendMessage

This call is used by the buyer to send a message concerning the return to the seller.

Sample Request

```
https://api.ebay.com/buy/order/v1/return/5000080362/sendMessage
```

Input Field/Container	Type	Occurrence	Description
returnId	long	Required	From returnId path parameter
message	String	Required	The body of the message is passed through this field

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
}
```

Output Container/Field	Type	Description
AckValue	String	Indicates the status of the call. All possible values SUCCESS,PARTIAL,FAILURE.
ResponseStatus	Integer	Commonly used status codes defined by HTTP, see Status Code List for the complete list.

POST /sell/order/v1/return/{returnId}/sendMessage

This call is used by the seller to send a message concerning the return to the buyer.

Sample Request

```
https://api.ebay.com/sell/order/v1/return/5000080362/sendMessage
```

Input Field/Container	Type	Occurrence	Description
returnId	long	Required	From returnId path parameter

Input Field/Container	Type	Occurrence	Description
message	String	Required	The body of the message is passed through this field

Sample Response

```
{
  "ackValue": "SUCCESS",
  "responseStatus": 200,
}
```

Output Container/Field	Type	Description
AckValue	String	Indicates the status of the call. All possible values SUCCESS,PARTIAL,FAILURE.
ResponseStatus	Integer	Commonly used status codes defined by HTTP, see Status Code List for the complete list.