Event Notifications Overview

Notifications are payloads of data that are provided as the order progresses through fulfillment stages, which can be sent to a JSON endpoint or custom application. These payloads are triggered automatically when any "event" happens in the fulfillment process, such as when an order is created or changes status. These events may originate from either the user interface or API – notifications are triggered by any appropriate action regardless of where it happens. Because these events occur at the site level, if a tenant-level action occurs (such as the creation of a new customer account) the system triggers the event notification for every site associated with the tenant.

The table in this guide lists some of the most common notifications, identifies the events that trigger them, and links to schemas detailing the parameters that are received in the payloads. These notifications are listed by "topics," or groups of notifications with related purposes.

Endpoint Configuration

For more information about subscribing custom applications to listen for events, refer to the Application Development documentation and Webhooks API documentation. These resources include technical guidelines for how the eventing process works, how to configure your endpoint, and schemas for each individual event. These configuration best practices are listed below.

> ℹ This endpoint can be up to a maximum of 250 characters.

Duplicate Prevention

Kibo expects a 200 OK success response to be returned from within 45 seconds of sending the notification. If no success response is received during that time or if an HTTP status error message is received, the notification will be placed in a queue to be resent every minute or so (the actual timing will vary based on the size of the queue). The notification will continue to be resent until either the success response is received or the notification expires after 14 days.

Due to this process, the endpoint may sometimes receive duplicates of notifications. To avoid saturation with a large number of duplicates, it is recommended to configure the endpoint with duplicate prevention.

Out-Of-Order Events

It is sometimes possible for notifications of different topics to be received out of order. While Kibo should always send notifications of the same topic in the order of the process flow, notifications from different topic groups are not guaranteed to be sent in any particular order. For instance, if a shipment event triggers both a Shipment Status and a Shipment Workflow notification, it is not known which will be received first by the endpoint. Thus, it is also advised to configure the endpoint to be capable of receiving out-of-order notifications to avoid any problems that may be caused by expecting these different notifications in a strict order.

Retry Behavior

It is also important to know that Kibo uses backoff behavior when re-queueing failed notifications. If there is a server outage or other issue preventing notifications from successfully sending, the system will retry several times. The first time the notification fails, there will be a five minute delay before attempting to send it again. The second time it fails, there will be a one hour delay. If it continues to fail, the amount of time between attempts will increase to the point of 24 hours before making the final attempt. If it still cannot go through, the notification will not be tried again.

This means that if orders are placed during a short-term outage that lasts a few hours, the notifications for those orders will eventually come through once the outage is resolved.

Event Notification Formatting

Notifications will always consist of limited event payload information that you can then use to retrieve additional information about that event. To version events, maintain security, and limit payload size, the event payload does not return the object itself… just the IDs.

For instance, when you receive an order.updated notification you can use the order ID in that payload to then make a GET call to the Order API and retrieve the full order data.

Notification Fields

Notification objects contain the following basic information to describe events. Some may provide additional details depending on the context, such as old and new shipment states or the reason that a change was made.

- `eventID` : A unique identifier for the event.
- `topic` : The type of event that occurred, such as product.created.
- `entityID` : Identifies the object for which the action occurred, such as a product code or order ID or shipment number.
- `timestamp` : The date-time at which the event occurred.
- `correlationID` : An identifier used by Kibo to track API requests across different services in the data logs.
- `isTest` : Boolean indicator that identifies whether the event was sent as part of a test.
- `extendedProperties` : Detailed metadata not provided by all notifications. This is a list of key:value string pairs, which vary depending on what information is being provided.

Notification Topics

Notifications are organized by topics, which are buckets of one or more notifications associated with a particular object such as orders or shipments.

Event Category	Entity ID Represents	Event	Description
Cart	Cart ID	cart.created	The first item was added to a new cart.
		cart.updated	The contents of a defined cart were updated.
		cart.deleted	A cart was deleted. The system deletes a cart when a shopper attempts to retrieve an expired cart, or when an anonymous shopper with an active cart logs in and merges the anonymous cart with the user's cart.
Category	Category ID	category.created	A new product category was created.
		category.updated	The properties of a defined product category were updated. This does not include event notifications when a client adds a product to or removes a product from a category.
		category.deleted	A defined product category was deleted.
		credit.created	A store credit or gift card was created for a customer account.

Event Category	Entity ID Represents	Event	Description
Credit	Credit ID	credit.updated	The properties of a store credit or gift card were updated. This includes notifications when a client associates the credit with a customer account and notifications when a shopper spends all or part of the credit balance.
		credit.deleted	A store credit or gift card was deleted.
Customer Account	Customer Account ID	customeraccount.created	A customer account was created in the tenant. Because customer accounts are stored at the tenant level, an event notification occurs for each site associated with the tenant.
		customeraccount.updated	The properties of a defined customer account were updated.
		customeraccount.deleted	A defined customer account was deleted.
		customersegment.created	A new customer segment was created.

Event Category	Entity ID Represents	Event	Description
Customer Segment	Customer Segment ID	customersegment.updated	The properties of a defined customer segment were updated. This does not include notifications for when a client adds customer accounts to or removes customer accounts from a customer segment.
		customersegment.customeradded	A customer account was added to the defined customer segment.
		customersegment.customerremoved	A customer account was removed from the defined customer segment.
		customersegment.deleted	A defined customer segment was deleted.
B2B Account	B2B Account ID	b2baccount.statuschanged	A B2B account changed its status. This is triggered when an account moves into any of the following states: ACTIVE, DISABLED, or PENDING APPROVAL.
		discount.created	A new discount or coupon was created.

Event Category	Entity ID Represents	Event	Description
Discount	Discount ID	discount.updated	The properties of a defined discount or coupon were updated. This does not include notifications when a shopper redeems a discount.
		discount.expired	A defined discount or coupon expired.
		discount.deleted	A defined discount or coupon was deleted.
Email	Email ID	email.requested	An email was requested.
Facet	Facet ID	facet.created	A new facet was created.
		facet.updated	A defined facet was updated.
		facet.deleted	A defined facet was deleted.
Inventory	Reservation ID	inventory.cartitemallocated	Inventory was successfully allocated for reservation.
	Cart ID	inventory.cartpendingitemcreated	Items were overallocated and a pending item was successfully created.
	Location Code	location.created	A new physical location was configured for a tenant environment.
		location.updated	One or more properties of a defined physical location are updated in the tenant environment.

Event Category	Code Entity ID Represents	Event	Description
Location		location.deleted	A defined physical location was deleted from the tenant environment.
	Location Type Code	locationtype.created	A new location type was defined used to group locations.
		locationtype.updated	One or more properties of a location type were updated, or locations were added to or removed from the location type.
		locationtype.deleted	A defined location type was deleted. You cannot delete locations types with associated locations.
		order.abandoned	A defined order was abandoned.
		order.addresschanged	An order's address was changed.
		order.cancelled	A defined order was cancelled.
		order.errored	An order experienced an error.
		order.closed	An order was closed after it was fully paid and fulfilled.

Event Category	Entity ID Represents	Event	Description
Order	Order ID	order.fulfilled	The order was shipped to the customer or an in-store pickup was created for fulfillment.
		order.imported	A historic order was imported.
		order.opened	A new order was accepted. This generally occurs after a successful checkout process. However, if you have order validation integrations installed, such as a fraud detection integration, this occurs after the order passes the review process.
		order.pendingreview	A defined order is pending further review.
		order.saved	An order was updated and saved.
		order.updated	One or more properties of a defined order was updated.
		return.cancelled	A defined return was cancelled.
		return.closed	The return was closed after a return replacement or refund was issued.

Event Category	Entity ID Represents	Event	Description
		return.opened	A return was created.
		return.updated	One or more properties of a defined return are updated.
		return.rejected	A return is marked as rejected.
	Payment ID	payment.authorized	An order payment was authorized.
		payment.captured	An order payment was captured. If both an authorization and a capture were performed at the same time, it is considered to be a capture event.
		payment.credited	An order payment was credited.
		payment.refunded	An order payment was refunded. Appeasements are also considered to be payment.refunded events.
		payment.transactionerror	An order payment experienced a transaction error.
		payment.voided	An order payment was voided.

Event Category	Entity ID Represents	Event	Description
		producttype.created	A new product type was created in any master catalog. The system triggers an event for each site that features the new product in its associated catalog.
		producttype.deleted	A defined product type was deleted.
		producttype.updated	One or more properties of a defined product type were updated.
		product.coderenamed	A defined product code was renamed.
		product.created	A new product was created in any master catalog. The system triggers an event for each site that features the new product in its associated catalog.
		product.updated	One or more properties of a defined product, product bundle, or a variation of a defined product were updated.
		product.deleted	A defined product was deleted.
Product	Product Code	productdraft.created	An update was made to a product definition in draft mode.

Event Category	Entity ID Represents	Event	Description
		productdraft.deleted	A product definition was deleted in a catalog associated with a master catalog that uses the "Pending" product publishing mode.
		productdraft.discarded	Draft changes to a product definition were discarded.
		productdraft.published	Draft changes to a product definition were published to a live site.
		productdraft.updated	One or more properties of a product definition were updated in an existing draft.
		productinventory.instock	A product previously out of stock returns to active inventory.
		productinventory.outofstock	The active inventory stock of a product at a location was depleted. The specific `locationCode` is included.
Reservation	Reservation ID	reservation.deallocated	An inventory reservation was deallocated.
		reservation.expired	An inventory reservation expired.
		return.authorized	A return was authorized.

Event Category	Entity ID Represents	Event	Description
Return	Return ID	return.cancelled	A return was cancelled.
		return.closed	A return was processed and closed.
		return.opened	A new return was created.
		return.rejected	A return was rejected instead of authorized.
		return.updated	A return was updated.
Shipment	Shipment ID	shipment.adjusted	An adjustment was made at the shipment level.
		shipment.itemadjusted	An adjustment was made at the shipment item level, such as when the item requirements are changed to make the shipment fulfillable by the assigned location.
		shipment.itemscanceled	A shipment line item was canceled.
		shipment.itemsdeclined	A shipment line item was declined.
		shipment.itemsrejected	A shipment line item was rejected.
		shipment.partialpickupready	At least one shipment line item is ready for customer pickup.

Event Category	Entity ID Represents	Event	Description
Shipment Status	Shipment ID	shipment.statuschanged	A shipment changed its status. This is triggered when a shipment moves into any of the following states: READY, REASSIGNED, BACKORDER, CANCELLED, FULFILLED, or CUSTOMER_CARE.
Shipment Workflow	Shipment ID	shipment.workflowstatechanged	A shipment changed its workflow state. While this may be related to shipment status events, this is not triggered by the status change itself. Instead, it is triggered by the shipment moving to another step in the fulfillment workflow (such as ACCEPTED_SHIPMENT or PRE_ACCEPT_SHIPMENT, depending on the implementation).
Search	Search Settings	searchsettings.updated	One or more search setting properties were updated.
		site.created	A new site was created.
		site.cloned	A defined site was cloned.

Event Category	Entity ID Represents	Event	Description
Site	Site ID	site.updated	One or more properties of a defined site were updated.
		site.deleted	A defined site was deleted.
Subscriptions	Subscription ID	subscription.statuschanged	A subscription changed status to Active, Paused, Cancelled, or Errored.
		subscription.activated	A subscription has been activated.
		subscription.cancelled	A subscription was cancelled.
		subscription.errored	A subscription experienced an error.
		subscription.paused	A subscription was paused.
		subscription.paymentupdated	A subscription's payment method was updated.
Tenant	Tenant ID	tenant.created	A new tenant was created.
		tenant.deleted	A defined tenant was deleted.
		tenant.expiring	A defined tenant is expiring.
		tenant.hard_deleted	A defined tenant was hard deleted from the system.

Event Category	Entity ID Represents	Event	Description
		tenant.updated	One or more properties of the defined tenant were updated.
		tenant.userlogin	A user logged in to a defined tenant.

Event Category	Entity ID Represents	Event	Description