

Feed API Playbook

A comprehensive integration guide for eBay's partners

Overview	2
About	2
Getting Started	3
Prerequisites	3
OAuth	3
Regional Constraints	3
Product Categories	4
Product Identifiers	5
Working with the Feed API	6
Scope	6
Filters	6
Nexus	6
Deployment Schedule	8
Methods	9
Implementation	9
Tracking	10
URL Structure	10
Examples	11

Overview

The playbook at hand shall provide EPN partners with a brief guideline on how to integrate with eBay's modern Feed API. It provides an overview of processual requirements, guidelines on how to use and integrate it and more.

About

The Feed API is by far the most resource-efficient interface to access the more than a billion eBay listings globally, as it allows partners to retrieve big chunks of structured listing data and regular updates with very few API calls.

It is designed as an alternative feed source to the Item Feed files provided as direct downloads via eBay's Partner Network platform. The new feeds provide major improvements in terms of accessibility, compatibility, and quality compared to the legacy Item Feeds.

Although the general format by how eBay listings are exposed has been retained due to the fundamental architecture of eBay's marketplace.

Below is an overview of the most important differences between the Feed API and the older EPN feeds:

	EPN Feed	Feed API
Download via EPN Portal	●	
Automatic Download via Interface	●	●
Download Files in Chunks		●
Full/Weekly Feed Available	●	●
Daily Feed Available	●	●
Hourly Feed Available		●
URLs Included	●	

Getting Started

The basic requirements to access the Feed API are as follows:

- Registered eBay Partner Network (EPN) publisher account - see <https://epn.ebay.com>
 - Registered eBay Developer Program (EDP) account - see <https://developer.ebay.com/api-docs/static/creating-edp-account.html>
 - Access request submitted to your account manager, containing:
 - EPN publisher account or campaignId
 - EDP account AppId & DevId of the respective keyset - see <https://developer.ebay.com/my/keys>
 - OAuth2.0 scope required - Feed API: https://api.ebay.com/oauth/api_scope/buy.item.feed
-

Prerequisites

OAuth

All eBay REST APIs use the OAuth 2.0 protocol for application and user authorization. You must provide a valid access token for each request you make to the eBay REST interfaces. OAuth access tokens verify to eBay that a request is coming from a valid application and that the application has the correct scope to carry out the requests.

More at <https://developer.ebay.com/api-docs/static/oauth-tokens.html>

Regional Constraints

The new Feed API doesn't support all eBay marketplaces, but we do steadily assess additional regions to add. Please have a look at the list of supported marketplaces to verify the suitability of the Feed API for your integration.

Supported marketplaces are listed at

<https://developer.ebay.com/api-docs/buy/static/ref-marketplace-supported.html>

Product Categories

To be able to fit eBay listings into your own taxonomy, it's highly recommended to implement a product category matching process. eBay's Taxonomy API provides you with access to individual category trees for each eBay marketplace.

eBay's product category tree consists of the following core elements relevant for a qualitative matching:

- categoryTreeld
 - identifies the respective marketplace
- categoryTreeVersion
 - identifies the current version and determines if you'll have to update the matching
- categoryId
 - a unique id that identifies a certain product category in the category hierarchy for a given marketplace
- categoryName
 - name of the respective category
- categoryTreeNodeLevel
 - category level within the complete category tree
- leafCategoryTreeNode
 - indicates if the respective category is located on the leaf level of the category tree

To fit a listing into your taxonomy you'll have to match eBay's current category tree structure, preferably by starting with the root or the L1 categories, through as many levels as possible. Every listing record provided by the Feed API provides a leaf categoryId, which can be located on any tree level between L2 and L_n.

This approach can help with certain integration approaches to augment one's product catalog independently of the availability of generic product identifiers, like GTIN or Brand/MPN. Furthermore, a proper taxonomy matching also helps to strengthen a product identifier based matching process, as the combination of category and productId can help to identify false-positive matchings.

More at <https://developer.ebay.com/api-docs/commerce/taxonomy/overview.html>

Product Identifiers

eBay tries to break all possible properties of a listing down to a single identifier, to be able to unambiguously identify the product sold through that eBay listing. That identifier is simply called eBay Product Identifier or EPID and it allows you to match a listing to a product that is already part of your taxonomy, with a high degree of accuracy.

It is highly recommended to use the EPID in conjunction with other product identifiers as it covers many more listings than GTIN or MPN/Brand identifiers do. It also allows you to speed up the feed import process for new listings whose EPID is already matched to your taxonomy.

Your current taxonomy should receive a horizontal extension on the product level to incorporate the EPID, as depicted in the following example:

Samsung Galaxy S9 SM-G960 - 64GB - Midnight Black

BRAND	MPN	GTIN_1	GTIN_2	EPID	EBAY_CATEG_ID
Samsung	SMG960UZXAXAA	0887276246529	NaN	243315735	9355
Samsung	SMG960UZXASPR	0887276245966	NaN	243389193	9355
Samsung	SMG960UZXAVZW	0887276244198	NaN	243306786	9355

Making use of the EPID is also recommended because it allows you to enrich your product catalog with product details provided by eBay.

Once an EPID is matched with one of your products, you won't have to try different identifiers, such as GTIN (EAN, UPC or ISBN) or Brand/MPN to access eBay's [Catalog API](#). Instead, you will always use the corresponding EPID to request additional product details.

It's also used for eBay's new [Marketing API](#) and allows for a product-based search on eBay's marketplaces with help of the [Browse API](#).

More on the mentioned APIs:

Catalog API - <https://developer.ebay.com/api-docs/commerce/catalog/overview.html>

Marketing API - <https://developer.ebay.com/api-docs/buy/marketing/overview.html>

Browse API - <https://developer.ebay.com/api-docs/buy/browse/overview.html>

Working with the Feed API

Scope

By default, data feeds provided by the Feed API do cover three different scopes:

- NEWLY_LISTED
 - All listings **added** to a certain L1 category on a specific date.
 - Also called “Daily (Incremental) Feed”
 - Updates available about every 24 hours
- ALL_ACTIVE
 - All listings **active** within a certain L1 category
 - Also called “Weekly Bootstrap Feed”
 - Updates available about every 7 days
- HOURLY
 - All listings **changed** within a certain L1 category through a specific hour
 - Also called “Hourly Snapshot Feed”
 - Updates available about every 60 minutes

Filters

Certain item filters are applied to all data feeds that cannot be altered by your application.

More details about the applied filters are available at

<https://developer.ebay.com/api-docs/buy/static/api-feed.html#feed-filters>

Nexus

The standard approach for implementing (see chap. [Implementation](#)) the Feed API follows a schema that covers all three [scopes](#). The combination is key for keeping listing details updated without having to rely on real-time APIs.

The coherence between the three scopes can be described as layers that complement each other.

- The weekly (bootstrap) feed provides the base or bootstrap of listings every week. **It can alternatively be imported just once to build a foundation of currently available items.**
- The daily (incremental) feed provides only new listings added on a given day during the week.
- The hourly (snapshot) feed provides updates to the bootstrap and daily feed.



Figure 1: Schema of the Feed APIs layered approach

Despite its complementary nature, the approach leaves the door open for a real-time extension based on eBay’s REST APIs.

It’s generally feasible to retrieve (near) real-time updates for items via eBay’s Browse API, but it’s highly recommended to use the Feed API as a foundation for that approach due to the very high number of listings available on eBay.



Figure 2: Schema of a Browse API real-time extension

Deployment Schedule

Combining all three scopes isn’t straightforward. All data feeds require processing time and their deployment schedule underlies a certain delay that differs for every scope.

- NEWLY_LISTED (Daily Feed)
 - Updates available about every 24 hours
 - Earliest availability typically at 09:00 AM MST (16:00 UTC)
 - Deployment delay typically 48 to 72 hours

- ALL_ACTIVE (Weekly Feed)

- Updates available about every 7 days
- Earliest availability typically on Wednesday 02:00 AM MST (09:00 UTC)
- Deployment delay typically 96 hours

- HOURLY (Hourly Feed)
 - Updates available about every 60 minutes
 - Earliest availability typically every full hour
 - Deployment delay typically 2 hours

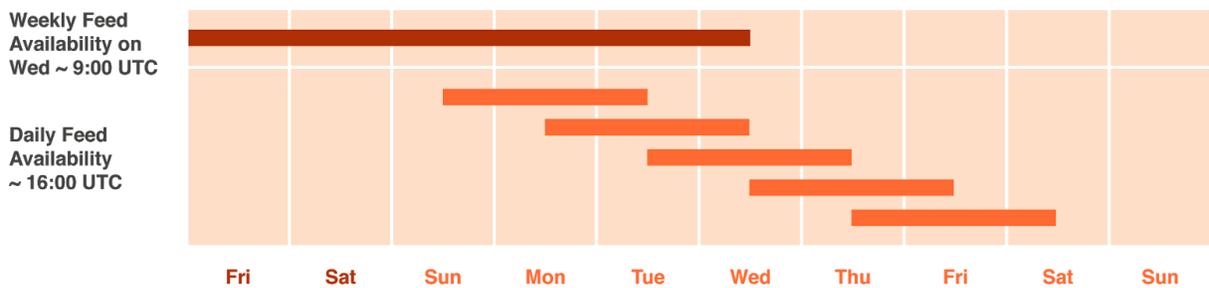


Figure 3: Schema of the Weekly and Daily Feed deployment delays

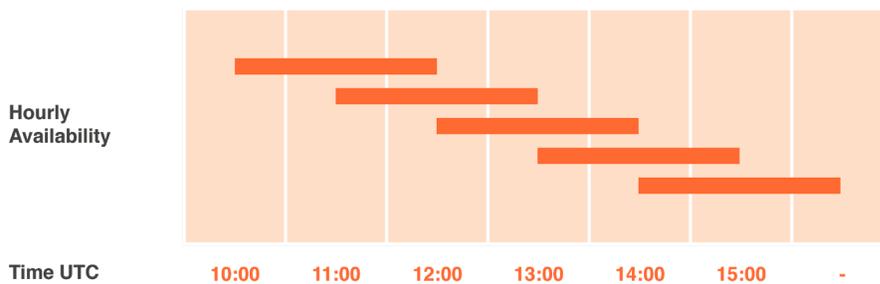


Figure 4: Schema of the Hourly Feed deployment delay

We do strongly advise you to implement your code to handle all delays generously, as they are approximate values and may vary under different circumstances.

Please note that dates and times are declared differently between the [getItemFeed](#) and the [getItemSnapshotFeed](#) methods when working with the Feed API.

The latter one is only accepting dates and times expressed according to [ISO 8601](#) and is based on UTC. To be precise, only the following formats are supported:

- YYYY-MM-DDThh:00:00.000Z

- YYYY-MM-DDThh:mm:ss
- YYYY-MM-DDThh:mm:ssZ

In comparison, the getItemFeed method requires just a simple “yyyyMMdd” format and is declared as MST (Mountain Standard Time).

Methods

The documentation of all supported Feed API methods can be found at:

<https://developer.ebay.com/api-docs/buy/feed/resources/methods>

It is best practice to combine all available methods for the purpose of integrity, currentness, and efficiency. Yet, all methods provide datasets that can be processed independently of the other methods for example by augmenting the data with details queried via [Browse API](#) and/or [Catalog API](#).

Implementation

All data feeds provided by the Feed API contain certain identifiers, like the mandatory itemId and categoryId or the optional GTIN/MPN or EPID, which are all fully compatible with eBay’s other RESTful APIs.

The Feed API, therefore, offers eBay’s partners a flexible integration path. It can be used as a standalone service or in combination with other resources, like our real-time APIs.

The eBay Developer Program is additionally offering a Java-based Feed API SDK available at Github:

<https://github.com/eBay/FeedSDK>

For more details on how to integrate with the Feed API please have a look at the comprehensive implementation guidelines available at

<https://developer.ebay.com/api-docs/buy/feed/static/overview.html>

Tracking

As EPN publisher you will want to advertise all eBay listings on your website with proper tracking links. Most EPN tools, and the REST APIs are capable of dynamically exposing individual tracking links for your EPN account.

However, the Feed API is not providing tracking links and you might have to construct them yourself.

URL Structure

When solely working with the Feed API, you’ll have to follow a certain schema, to make your eBay advertisements trackable.

1. Rover

a. All tracking links point to the same primary URL structure:
[https://rover.ebay.com/rover/1/\[ROTATION-ID\]/1?campid=\[EPN-CAMPAIGN-ID\]&customid=\[SUBID\]&toolid=20006&mpre=\[TARGET\]](https://rover.ebay.com/rover/1/[ROTATION-ID]/1?campid=[EPN-CAMPAIGN-ID]&customid=[SUBID]&toolid=20006&mpre=[TARGET])

b. Parameters:

[ROTATION-ID]	The rotation Id fitting the [TARGET] marketplace
[EPN-CAMPAIGN]	Your individual EPN campaign Id
[SUBID]	Your optional sub Id
[TARGET]	The target URL (see next

c. List of relevant marketplaces and respective rotation Ids:

- eBay US 711-53200-19255-0
- eBay CA 706-53473-19255-0
- eBay ES 1185-53479-19255-0
- eBay FR 709-53476-19255-0
- eBay IT 724-53478-19255-0
- eBay UK 710-53481-19255-0
- eBay NL 1346-53482-19255-0
- eBay DE 707-53477-19255-0
- eBay AT 5221-53469-19255-0
- eBay CH 5222-53480-19255-0

Please note:

The Feed API is currently only providing items of certain marketplaces, listed under [Regional Constraints](#). Many items listed on those marketplaces are eligible for cross border trade and can be shipped to customers from other countries though.

In case your customers are located in a country listed above, make use of the corresponding top-level domain and rotation Id to redirect them to their local eBay marketplace.

2. Target

a. The target URL is the secondary (URL-encoded) URL structure and replaces the [TARGET] placeholder of the primary rover URL
[https://www.ebay.\[TLD\]/itm/\[LISTING-ID\]?var=\[VARIATION-ID\]](https://www.ebay.[TLD]/itm/[LISTING-ID]?var=[VARIATION-ID])

b. Parameters:

[TLD]	The top-level domain of the respective marketplace
[LISTING-ID]	The middle part of the itemId → v1 111110541594 410188579148
[VARIATION-ID]	The last part of the itemId → V1 111110541594 410188579148

Examples

- [EPN-CAMPAIGN] = 1234567890
- [SUBID] = abc123!

itemId	Market place	Target	Rover
v1 302266213227 601108617967	eBay DE	https://www.ebay.de/itm/302266213227?var=601108617967	https://rover.ebay.com/rover/1/707-53477-19255-0/1?campid=1234567890&customid=abc123!&toolid=10050&mpre=https%3A%2F%2Fwww.ebay.de%2Fitm%2F302266213227%3Fvar%3D601108617967
v1 163590899843 0	eBay UK	https://www.ebay.co.uk/itm/163590899843?var=0	https://rover.ebay.com/rover/1/710-53481-19255-0/1?campid=1234567890&customid=abc123!&toolid=10050&mpre=https%3A%2F%2Fwww.ebay.co.uk%2Fitm%2F163590899843%3Fvar%3D0
v1 183591580738 690989461642	eBay US	https://www.ebay.com/itm/183591580738?var=690989461642	https://rover.ebay.com/rover/1/711-53200-19255-0/1?campid=1234567890&customid=abc123!&toolid=10049&mpre=https%3A%2F%2Fwww.ebay.com%2Fitm%2F183591580738%3Fvar%3D690989461642