



New GraphQL APIs

July 2026
eBay Connect

Ramesh Ramachandran

Why Migrate?

eBay is modernising its entire platform API surface

3

Legacy API Families
Being Replaced

2

New Technology Stacks
REST + GraphQL

7+

New API Families

100%

Sell + Buy Flows
Covered

What's Going Away

Trading API (SOA / XML-based)
Post-Order API
Legacy Platform Notifications

What's Replacing It

GraphQL APIs (new,
purpose-built)
REST APIs (expanded coverage)
Modern event/notification model

Key Benefits

Request only the data you need
Consistent OAuth 2.0 auth
Better developer experience

GraphQL Operations: 91 (34 queries, 57 mutations)

Listing Management APIs

Traditional AddItem/Reviseltem/EndItem family → Listing API

Order Management APIs

GetOrders / GetSellerTransactions → Order API + Item Commitment API

Key Concept:

Orders are now split into ORDERS (paid) and ITEM COMMITMENTS (unpaid). TransactionID is replaced by orderLineItemId.

Post-Order: Cancellations

Post-Order Cancellation API → Cancellation API

Buyer-facing operations NOT migrated.

Post-Order: Order Inquiries (INR)

Item Not Received inquiries & cases → Order Inquiry API

Post-Order: Returns

Post-Order Return API → Return API

Separate Case entity eliminated.

Why GraphQL ?

No Over-fetching

Clients request only specific fields. This prevents downloading unnecessary data and saves network bandwidth.

No Under-fetching

Clients request nested, related resources simultaneously, eliminating the need for multiple successive API calls.

Single Endpoint

All operations run through one URL (via HTTP POST). This simplifies backend routing and client configuration.

Strongly-Typed Schema

A strict schema acts as a contract between client and server, catching data errors early during development.

Built-in Introspection

The API is inherently self-documenting. Developers can query the schema directly to discover available fields.

Smooth API Evolution

Fields can be added or deprecated gradually, eliminating the need for strict API versioning like /v1/ or /v2/.

Native Real-Time Support

It features built-in "Subscriptions" which allow out-of-the-box data streaming over WebSockets.

GraphQL Explorer Demo

The screenshot shows the GraphQL Explorer interface for the eBay Developers Program. The browser address bar displays the URL: `developer.ebay.com/my/graphql_explorer?index=0&env=production`. The page header includes the eBay logo, "Developers Program", and navigation links: "Join", "Develop", "Grow", "Updates", "Support", and "Hi rameshpoti". The main heading is "GraphQL Explorer". Below this, the environment is set to "Production" (indicated by a selected radio button). A dropdown menu shows "ramesh-test". The interface is divided into three main sections: Explorer, Query Editor, and Results.

Explorer: A tree view on the left lists various GraphQL endpoints. The "sellerListings" endpoint is selected and expanded, showing sub-queries like "sellerListingsSearch" and "sellerListings".

Query Editor: The central area displays a GraphQL query for "SellerListingsItemTitle". The query is as follows:

```
1 query SellerListingsItemTitle {
2   sellerListings(input: {listings: {listin
3     ... on ListingsSuccess {
4       __typename
5       listings {
6         ... on ListingSuccess {
7           __typename
8           listing {
9             listingId
10            sellerProduct {
11              title
12            }
13          }
14        }
15      }
16    }
17  }
18 }
```

Results: The right panel shows the JSON response for the query. The response is a nested object with "data" containing "sellerListings", which in turn contains "listings". The "listings" array has one element, "ListingSuccess", which contains a "listing" object. The "listing" object has "listingId" and "sellerProduct" fields. The "sellerProduct" object has a "title" field with the value "New Balance MF791MM Men's Mid Top Lace-Up Plastic Cleats Sneaker White/Silver".

Variables: The bottom panel shows the variables defined for the query. The variables are:

```
1 {
2   "Authorization": "Bearer v^1.1#1^1#1^3#r^0#f^0"
3 }
```



Thank you