



Web Services REST API Guide

Version 3.7



Copyright Notice

© 2004–2007 eBay Inc. All rights reserved.

The information contained in this software manual is proprietary and confidential. For information, contact Developer Relations:

developer-relations@ebay.com

This manual and the REST API may only be used in accordance with the terms of the Developers Program agreement.

References to names of people and companies in source code examples or descriptions are for demonstration purposes only and do not refer to any real people or companies.

eBay and the eBay logo are registered trademarks of eBay Inc. All other brands are the property of their respective owners.

Contents

Chapter 1: Using the REST API	1
Introductory Samples	1
Information in this Guide	2
Calls Available in the REST API	2
Basic REST Input Parameters	3
Root URLs, Containers, and Repeating Parameters	3
Required Parameters	4
Error Information	5
Affiliate Parameters	5
Specifying International Sites	7
Obtaining Tokens and Testing Calls	7
Obtaining a REST Token for the Sandbox	8
Using the REST API in Production	8
Testing REST Calls Using a Browser	9
Find Items with GetSearchResults	9
GetSearchResults Samples	10
GetSearchResults Input Parameters	10
GetSearchResults Query Modifiers	12
GetSearchResults Output Fields	12
Find Items in a Category with GetCategoryListings	13
Find Items with GetSearchResultsExpress	14
GetSearchResultsExpress Samples	14
Retrieve Product Information with GetProducts	15
Obtain Item Data with GetItem	17
Obtain Shipping Costs with GetItemShipping	18
Obtain a Category Hierarchy with GetCategories	18
Obtain Keywords with GetPopularKeywords	19
Transforming Data Returned	20
Uploading and Using XSL Files	20
Viewing, Renaming, and Deleting XSL Files	21
Legacy REST GetSearchResults Input	23
Legacy Samples for GetSearchResults	23
Legacy Affiliate Parameters for GetSearchResults	23
Legacy Input Parameters for GetSearchResults	24
Chapter 2: Additional Resources	29
REST Information Sources	29



Index	eBay API Documentation	29
		30

Chapter 1

Using the REST API

With the eBay REST API, you can use an URL for simple or complex searches of millions of item listings. Moreover, you can use an URL to get detailed information about individual item listings.

Most calls to the REST API are from a Web page, application, or script. You can use the REST API to integrate eBay functionality into your Web sites and applications. For the calls available in the REST API, please see “[Calls Available in the REST API](#)” on page 2.

Note: The REST API can receive unified schema input parameters for `GetSearchResults`. The samples in this document have been updated to show the unified schema input parameters (except in the section “[Legacy REST GetSearchResults Input](#)” on page 23). For information about using unified schema input parameters for `GetSearchResults`, please see “[Find Items with GetSearchResults](#)” on page 9.

The System Announcements page is the primary resource for checking the status of issues relating to the Sandbox and the Production servers. Please check the System Announcements page before reporting urgent production system issues to eBay. Typical issues we announce on this page are latency, unavailability, known errors, and known call failures.

eBay Developers Program System Announcements Forum

<http://developer.ebay.com/forums>

General information about the unified schema, on which the REST API is based, is available at the following location:

<http://developer.ebay.com/support/docs/>

To learn about the eBay Affiliate Program, please see <http://affiliates.ebay.com>. If this link or other links in this guide do not function, please close any open browser windows before using them. For information about affiliate parameters in call requests, see “[Affiliate Parameters](#)” on page 5.

Introductory Samples

This sample shows how to use a **Query** parameter to search eBay:

```
http://rest.api.ebay.com/  
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=toy  
%20boat&Version=491&UnifiedInput=1
```

This sample shows how to use a **Query** parameter to search eBay Express:

```
http://rest.api.ebay.com/  
restapi?CallName=GetSearchResultsExpress&RequestToken=UserToken&RequestUserId=UserName&Que  
ry=garmin+gps&EntriesPerPage=3&Version=491
```

Information in this Guide

For information about obtaining a string for the `RequestToken` parameter, see [“Obtaining Tokens and Testing Calls”](#) on page 7. For parameters required with all REST calls, see [“Required Parameters”](#) on page 4. For additional samples, see the call sections of this guide (such as [“Find Items with GetSearchResults”](#) on page 9).

This guide contains the following sections:

- [“Calls Available in the REST API”](#) on page 2
- [“Basic REST Input Parameters”](#) on page 3
- [“Obtaining Tokens and Testing Calls”](#) on page 7
- [“Find Items with GetSearchResults”](#) on page 9
- [“Find Items in a Category with GetCategoryListings”](#) on page 13
- [“Find Items with GetSearchResultsExpress”](#) on page 14
- [“Retrieve Product Information with GetProducts”](#) on page 15
- [“Obtain Item Data with GetItem”](#) on page 17
- [“Obtain Shipping Costs with GetItemShipping”](#) on page 18
- [“Obtain a Category Hierarchy with GetCategories”](#) on page 18
- [“Obtain Keywords with GetPopularKeywords”](#) on page 19
- [“Transforming Data Returned”](#) on page 20
- [“Legacy REST GetSearchResults Input”](#) on page 23
- [“REST Information Sources”](#) on page 29

Calls Available in the REST API

For the REST parameters applicable to all or most calls, see [“Required Parameters”](#) on page 4.

The calls available for use in the REST API are the following:

- **GetSearchResults.** To search eBay, see [“Find Items with GetSearchResults”](#) on page 9.
- **GetCategoryListings.** To find items on eBay based on their category, see [“Find Items in a Category with GetCategoryListings”](#) on page 13.
- **GetSearchResultsExpress.** To search eBay Express, see [“Find Items with GetSearchResultsExpress”](#) on page 14.
- **GetItem.** To obtain detailed information about an item, see [“Obtain Item Data with GetItem”](#) on page 17.
- **GetItemShipping.** To obtain estimated shipping cost information for an item, see [“Obtain Shipping Costs with GetItemShipping”](#) on page 18.
- **GetCategories.** To retrieve the latest category hierarchy, see [“Obtain a Category Hierarchy with GetCategories”](#) on page 18.
- **GetPopularKeywords.** To obtain keywords that users have frequently specified when searching eBay, see [“Obtain Keywords with GetPopularKeywords”](#) on page 19.

Basic REST Input Parameters

This section of the REST API Guide contains the following subsections:

- “Root URLs, Containers, and Repeating Parameters” on page 3
- “Required Parameters” on page 4
- “Error Information” on page 5
- “Affiliate Parameters” on page 5
- “Specifying International Sites” on page 7

Root URLs, Containers, and Repeating Parameters

This section contains the following subsections:

- “Root URLs” on page 3
- “Containers” on page 3
- “Repeating Parameters” on page 4

Root URLs

To make a call to the REST API in the production environment (which contains live eBay data), use the following root URL:

<http://rest.api.ebay.com/restapi>

Additionally, if a call is supported in the Sandbox (test) environment, you can make a test call to the following root URL:

<http://rest.api.sandbox.ebay.com/restapi>

Here is an example of a GetSearchResults request that uses the REST API. The value of **Query** is “toy boat” with the space character HTTP-encoded as %20:

<http://rest.api.ebay.com/restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=toy%20boat&Version=491&UnifiedInput=1>

The overall size of the string after the “?” in the URL must be no larger than 2071 characters. If you access the REST API through a proxy, the proxy may place further size limitations on the URL.

Containers

Input parameters in the REST API can contain other parameters and can have attributes.

Note: In GetSearchResults, if an input parameter is unique among all of the possible input parameters for the call, then you do not need to specify the parameter container.

You can use XPATH to specify parameters that are containers or have attributes.

- **Nested Parameters.** Use the “.” symbol (i.e., Parent.Child).
- **Attributes.** Use the “@” symbol (i.e., Element@attribute).

Repeating Parameters

Beginning in early 2007, if parameters in a request repeat, you can use the “()” symbol, i.e., Element(Number). Before early 2007, parameters in a request cannot repeat.

For example, the following URL contains parameters for including descriptions in a search and also for specifying that only items listed with Now and New be returned:

```
http://rest.api.ebay.com/
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=slippers&SearchFlags(1)=SearchInDescription&SearchFlags(2)=NowAndNew&Version=491&UnifiedInput=1
```

Please note that if you are using the GetPopularKeywords call, you cannot specify a **CategoryID** parameter more than once per call.

Required Parameters

Several input parameters, listed in the table below, are required in REST API calls. Additional information about required REST input parameters is in the Knowledge Base:

Knowledge Base

<http://developer.ebay.com/DevZone/support/knowledgebase.asp>

For specifying a site other than the US site (which is site 0) in your call, include a `SiteId` parameter for the site. For more information, please see “[Specifying International Sites](#)” on page 7.

To learn about the eBay Affiliate Program, please see <http://affiliates.ebay.com> and “[Affiliate Parameters](#)” on page 5.

The following table describes the required input parameters.

Table 1-1 Required REST Input Parameters

Parameter	Description
<code>CallName</code>	The name of the API call.
<code>RequestToken</code>	The REST token of an eBay user. This token corresponds to a specific eBay user. See “ Obtaining a REST Token for the Sandbox ” for information on how to retrieve this value.
<code>RequestUserId</code>	The user name of the eBay user whose REST token is specified in the <code>RequestToken</code> parameter. The <code>RequestUserId</code> parameter is mandatory.
<code>UnifiedInput</code>	<p>The <code>UnifiedInput</code> parameter applies only to <code>GetSearchResults</code>.</p> <p>For <code>GetSearchResults</code>, if you include this parameter and set it to 1, it indicates you are using unified schema input. For more information, see “Find Items with GetSearchResults” on page 9.</p> <p>If you include the <code>UnifiedInput</code> parameter (and set it to 1), you do not need to specify the <code>Schema</code> parameter. If you do not include the <code>UnifiedInput</code> parameter (and set it to 1), you must include the <code>Schema</code> parameter and set it to 1.</p>

Table 1-1 Required REST Input Parameters

Parameter	Description
Version	<p>Specifies the API version you are using for the call. If the data returned is different from expected, test your call using a higher version, such as the minimum of 475 for GetSearchResultsExpress.</p> <p>The Version parameter replaces the CompatibilityLevel parameter; the CompatibilityLevel parameter is deprecated.</p> <p>If you do not use a Version parameter (or, for GetSearchResults legacy REST input, a CompatibilityLevel parameter), various problems can occur. Recent API versions have more features than older versions, and version thresholds are periodically set as the minimum supported version. More information is in the unified schema documentation accessible from http://developer.ebay.com/support/docs/.</p>
Schema	<p>The Schema parameter only applies to GetSearchResults.</p> <p>For GetSearchResults, if you do not include the UnifiedInput parameter (and set it to 1), you must include the Schema parameter and set it to 1.</p> <p>Thus, the Schema parameter is required only in the following case: In GetSearchResults, you are not using the UnifiedInput parameter. In such a case, the Schema=1 parameter returns data in the unified schema format. (If omitted in this case, or if you set it to 0, the results are returned in the legacy XML API format, which is <i>not</i> supported as of June 1, 2006.) For additional information, see “Find Items with GetSearchResults” on page 9.</p>

Error Information

The REST API returns error information if a call is invalid. Error information is returned within an **Errors** field. The **Errors** field includes a **ShortMessage** field and a **LongMessage** field. The **ShortMessage** field indicates why the error occurred. The **LongMessage** field provides a more verbose message, if applicable. For additional information, see the unified schema documentation, on which the REST API is based:

<http://developer.ebay.com/support/docs/>

Affiliate Parameters

If you use affiliate parameters, it is possible to obtain affiliate commissions based on calls made by your Web page, application, or script. Affiliate parameters enable the tracking of user activity. For more information about the eBay Affiliate Program, please see <http://affiliates.ebay.com>. For individual descriptions of the affiliate parameters, see “[Affiliate Parameter Descriptions](#)” on page 6.

Commissions are generated when users submit bids or do other qualifying actions. For example, if a user bids on an item after clicking a search result, a commission is generated.

In the REST API, affiliate parameters can be used with the following calls:

- `GetSearchResults` (see “Find Items with `GetSearchResults`” on page 9)
- `GetSearchResultsExpress` (see “Find Items with `GetSearchResultsExpress`” on page 14)
- `GetProducts` (see “Retrieve Product Information with `GetProducts`” on page 15)

This sample shows how to specify affiliate parameters in `GetSearchResults`:

```
http://rest.api.ebay.com/
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=tape
&TrackingPartnerCode=MyTrackingPartnerCode&AffiliateUserID=MyAffiliateUserID&TrackingID=MyTrackin
gID&Version=491&UnifiedInput=1
```

This sample shows how to specify affiliate parameters for `GetSearchResultsExpress`:

```
http://rest.api.ebay.com/
restapi?CallName=GetSearchResultsExpress&RequestToken=UserToken&RequestUserId=UserName&Que
ry=tape&TrackingPartnerCode=MyTrackingPartnerCode&AffiliateUserID=MyAffiliateUserID&TrackingID=M
yTrackingID&Version=491
```

If you use affiliate parameters with `GetSearchResults` or `GetSearchResultsExpress`, the REST API returns a modified version of the `ViewItemURL` string. The modified version of the `ViewItemURL` string includes information for tracking user activity.

Affiliate Parameter Descriptions

This section contains a table that describes the affiliate parameters.

Table 1-2 Parameters for Affiliate Tracking

Parameter	Required?	Data Type	Purpose
<code>TrackingPartnerCode</code>	N	Integer	<p>Specifies the third party, e.g. Commission Junction, who is your tracking partner (tracking provider).</p> <p>Depending on your tracking partner, specify one of the following values. Not all partners are valid for all sites.</p> <ul style="list-style-type: none"> 1 = Commission Junction 2 = Be Free 3 = Affilinet 4 = TradeDoubler 5 = Mediaplex 6 = DoubleClick 7 = Allyes 8 = BJMT

Table 1-2 Parameters for Affiliate Tracking

Parameter	Required?	Data Type	Purpose
TrackingID	N	String	Specifies an ID to identify you to your tracking partner (tracking provider). The value you specify is obtained from your tracking partner. For example, in the case of Commission Junction, the TrackingID is the PID given to you by Commission Junction.
AffiliateUserID	N	String	Optional value that depends on your tracking partner (tracking provider). For example, if you specify a TrackingPartnerCode of 1 for Commission Junction, the AffiliateUserID is your Commission Junction Shopper ID (SID).

Specifying International Sites

To specify a site other than the US site (which is site 0) in your call, include a `SiteId` parameter for the site and, for `GetSearchResults`, the `SearchLocationFilter.SearchLocation.SiteLocation.SiteID` parameter. The `SiteId` parameter is case sensitive.

For example, to search for items listed on the Germany site, you could specify the following:

```
http://rest.api.ebay.com/
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=toy
%20boat&Version=491&UnifiedInput=1&SiteId=77&SearchLocationFilter.SearchLocation.SiteLocation.Sit
eID=SiteImplied
```

Please see the “Field Differences for eBay Sites” topic in the unified schema documentation for additional information about international sites, including the values to use in the `SiteId` parameter:

```
http://developer.ebay.com/DevZone/XML/docs/WebHelp/
index.htm?context=eBay_XML_API&topic=FieldDifferences
```

Obtaining Tokens and Testing Calls

This section of the REST API Guide contains the following subsections:

- “Obtaining a REST Token for the Sandbox” on page 8
- “Using the REST API in Production” on page 8
- “Testing REST Calls Using a Browser” on page 9

Obtaining a REST Token for the Sandbox

To begin testing a REST API application in the Sandbox environment (“the Sandbox”) you must use the following steps to obtain a REST token. Note that `GetSearchResults` is supported in the Sandbox, but `GetSearchResultsExpress` is supported only in the production environment:

- 1 Join eBay by registering for an eBay User ID.
- 2 Join the eBay Developers Program. Information on joining is available at the following location. After you join, note the Sandbox keys you receive via email.

<http://developer.ebay.com>

- 3 Create a Sandbox environment test user with the Sandbox User Registration Tool at the following location. The Sandbox User Registration Tool creates a test user for the Sandbox test environment. Note the user name and password of the test user.

<http://developer.ebay.com/DevZone/sandboxuser.asp>

- 4 With the Sandbox keys you noted in Step 2, and with the user name and password of the test user you noted in Step 3, obtain a REST token for the Sandbox at the following location.

<http://developer.ebay.com/tokenool>

The REST token is used by eBay to identify who makes calls to the REST API. The REST token must be specified in the **RequestToken** parameter in your REST API calls. You can use the `GetUser` call, in the unified schema, to obtain a user’s REST token if one is available. For information about `GetUser`, see the following topic:

[GetUser](#)

- 5 Congratulations! You are ready to begin making REST calls in the Sandbox environment. See the Sandbox URL for the REST API in “[Basic REST Input Parameters](#)” on page 3.

Note: The REST token is not the same as the authentication token required to use other eBay APIs.

Using the REST API in Production

After testing your REST calls in the Sandbox environment (to get set up in the Sandbox environment, see “[Obtaining a REST Token for the Sandbox](#)” on page 8), you are ready to use the production environment.

To use the REST API in the production environment, you must obtain production keys. The Sandbox keys described in “[Obtaining a REST Token for the Sandbox](#)” on page 8 do not apply to the production environment.

To use the REST API in the production environment:

- 1 Join eBay by registering for an eBay User ID.
- 2 Join the eBay Developers Program.
- 3 Submit your application or Web page for certification and note your production keys. Information on certification is available at the following location.

<http://developer.ebay.com>

- 4 Using your production keys, obtain a REST token for an eBay user at the following location:
<http://developer.ebay.com/tokentool>

If you already have obtained a REST token for an eBay user, you can retrieve it with the GetUser call. For more information, see GetUser in the unified schema documentation, accessible from the following location:

<http://developer.ebay.com/support/docs/>

- 5 To begin making REST calls in production, see the production URL for the REST API in “Basic REST Input Parameters” on page 3.

Testing REST Calls Using a Browser

You can make calls to the REST API using a Web page or an application. You also can specify an URL in a Web browser.

A Web browser displays the XML returned by the API. If you have uploaded and specified an XSL file (see “Transforming Data Returned” on page 20), a browser may display search results in HTML.

You can do a test call in the Sandbox environment using a browser. Before you do so, follow the instructions in “Obtaining a REST Token for the Sandbox” on page 8. You can make the test call using the following URL. Substitute your Sandbox token and Sandbox user values for the `UserToken` and `UserName` values. The token you use must correspond to the username you use.

[http://rest.api.sandbox.ebay.com/
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=toy
%20boat&Version=491&UnifiedInput=1](http://rest.api.sandbox.ebay.com/restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=toy%20boat&Version=491&UnifiedInput=1)

Data is returned as an XML document.

Find Items with GetSearchResults

With GetSearchResults, you can use keywords and many other criteria to retrieve items listed on eBay. Rich functionality makes GetSearchResults a powerful call for retrieving items.

Also see the section “Basic REST Input Parameters” on page 3 for information about URLs used and required parameters.

Formerly, REST input parameters for GetSearchResults were similar to the legacy XML API input parameters for GetSearchResults. Now, you must begin specifying unified schema parameters for GetSearchResults. The unified schema is standardized and uses objects for data exchange. Additionally, parameters specific to GetSearchResults are not case sensitive.

All applications must be migrated from legacy input parameters to unified schema input parameters. For more information, see “GetSearchResults Input Parameters” on page 10.

Using `GetSearchResults`, you can filter results using many criteria, including category and currency. For example, an URL similar to the following might be used to search a category (617) for listings with “star+trek”:

```
http://rest.api.ebay.com/  
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&CategoryID  
=617&Query=star+trek&Version=491&UnifiedInput=1
```

The keywords can include wildcards. The response is an array. You can filter the item listings returned using such criteria as the listing category, the location, whether the item is listed in an eBay Store, and the listing type.

You cannot use `GetSearchResults` to search or browse eBay Express. For information about finding items on eBay Express, see “[Find Items with GetSearchResultsExpress](#)” on page 14.

The following sections describe specific aspects of the `GetSearchResults` call:

- “[GetSearchResults Samples](#)” on page 10
- “[GetSearchResults Input Parameters](#)” on page 10
- “[GetSearchResults Query Modifiers](#)” on page 12
- “[GetSearchResults Output Fields](#)” on page 12

GetSearchResults Samples

This section contains REST API samples that use the `GetSearchResults` call.

Please substitute your eBay user name and your REST token for `UserName` and `UserToken`.

The following URL provides a search of the listing title and description using “fuzzy slippers”; a maximum of 5 results will be returned:

```
http://rest.api.ebay.com/  
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=fuzz  
y%20slippers&EntriesPerPage=5&PageNumber=1&SearchFlags=SearchInDescription&Version=491&Unifi  
edInput=1
```

The following URL uses a **Query** parameter of “kitchen table” and sorts the items in the response in ascending order based on whether PayPal is accepted (accepted followed by unaccepted):

```
http://rest.api.ebay.com/  
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=kitch  
en%20table&Order=SortByPayPalAsc&Version=491&UnifiedInput=1
```

GetSearchResults Input Parameters

Formerly, REST input parameters for `GetSearchResults` were similar to the legacy XML API input parameters for `GetSearchResults`.

Now, all applications must be migrated from legacy input parameters to unified schema input parameters. The unified schema is standardized and uses objects for data exchange.

Please make use of the unified schema input format and begin migrating all applications from legacy input parameters to unified schema input parameters. To use the unified schema input format, include the **UnifiedInput** parameter and set it to 1 (**UnifiedInput=1**), as in the examples in this guide. Descriptions of the **UnifiedInput** parameter and the related **Schema** parameter are in the following sections: “[The UnifiedInput and Schema Parameters for GetSearchResults](#)” on page 11 and “[Required Parameters](#)” on page 4.

Input parameters for GetSearchResults are the same in the REST API as in the unified schema, except for the REST-specific input parameters. For the REST-specific input parameters, see “[Basic REST Input Parameters](#)” on page 3. For more information about using GetSearchResults, including descriptions of the call-specific input and output parameters, see the following unified schema document:

[GetSearchResults](#)

General information about the unified schema format is available at the following location:

<http://developer.ebay.com/support/docs/>

The following location contains additional information about using GetSearchResults in the unified schema, as well as links to related topics:

http://developer.ebay.com/DevZone/XML/docs/WebHelp/index.htm?context=eBay_XML_API&topic=SearchingItemsWithQuery

The UnifiedInput and Schema Parameters for GetSearchResults

For GetSearchResults, to use unified schema input, include the **UnifiedInput** parameter and set it to 1. All applications must be migrated from legacy input parameters to unified schema input parameters.

If you do not include the **UnifiedInput** parameter (and set it to 1), you must include the **Schema** parameter and set it to 1. The **Schema** parameter has the following characteristics:

- **Schema** applies only to GetSearchResults.
- **Schema** sets the output format to the unified schema format.
- If the **UnifiedInput** parameter is omitted (or set to 0), and **Schema** is omitted or set to 0, output is in the Legacy XML API format (unsupported as of June 1, 2006).

If you specify **UnifiedInput=1** in your input, then the **Schema** parameter is unnecessary; the output format automatically will be in the unified schema format.

GetSearchResults Query Modifiers

You can use the following modifiers in a GetSearchResults query:

Table 1-3 GetSearchResults Query Operators and Punctuation

Modifier	Purpose	Example of Usage in REST API	Returns
Space between words	Applies AND logic to multiple keywords.	<code>baseball%20card</code>	Items with both the words "baseball" and "card".
Comma between words, no parentheses	Applies AND logic to multiple keywords.	<code>baseball,card</code>	Items with both the words "baseball" and "card".
Comma between words in parentheses	Applies OR logic to multiple keywords.	<code>(baseball,card)</code>	Items with either the word "baseball" or the word "card".
Quoted words	Requires an exact sequence of words.	<code>"baseball%20card"</code>	Items with the exact phrase "baseball card". Note that spaces must be URL-encoded as %20.
-word	Specified word cannot be present.	<code>baseball%20-autograph</code>	Items that have the word "baseball" but not "autograph"
-(word, word, word)	Specified words cannot be present.	<code>baseball%20-(autograph,card,star)</code>	Items with the word "baseball" but not "autograph," "card," or "star".
*	Substitutes for one or more characters.	<code>baseball*</code>	Items starting with the string "baseball".
@ sign (URL-encoded as %40)	Search must find two of three words from a list.	<code>%40%20baseball%20autograph%20card</code>	Items with two of the three words "baseball," "autograph," and "card". For a "3 out of 4" search, use @2 (URL-encoded as %402) and a list of four words.
+	Concatenates a query string using AND logic with another keyword.	<code>%40%20baseball%20autograph%20card%20+star</code>	Items with any two of the three words "baseball," "autograph," or "card" in the title plus the word "star".

GetSearchResults Output Fields

The output of the GetSearchResults call in the REST API is similar to the output of the GetSearchResults call in the unified schema. Please see the following unified schema topic for the latest GetSearchResults output fields:

http://developer.ebay.com/DevZone/XML/docs/Reference/eBay/io_GetSearchResults.html#Response

Information about pagination is in the following section: [“Paginating Search Results”](#) on page 13.

After you have tested your program’s ability to handle output, you can use your program in production to transform output or you can enable server-side processing. Server-side processing uses your XSL files to format the output for you. For more information on server-side processing that uses your XSL files, see [“Transforming Data Returned”](#) on page 20.

Paginating Search Results

GetSearchResults returns items in “pages” to reduce the amount of data sent to clients in response to any one call.

For pagination, you use the **EntriesPerPage** and **PageNumber** input parameters, as in the example with these parameters at “[GetSearchResults Samples](#)” on page 10. **EntriesPerPage** is the maximum number of **ItemType** objects that should be returned in the **SearchResultItemArray** for any given call. **PageNumber** is the number of the page of data to return in the current call (default is 1).

In the response, **ItemsPerPage** indicates the number of items returned by the current call. **PaginationResult.TotalNumberOfEntries** indicates the total number of items matching the search criteria. **PaginationResult.TotalNumberOfPages** indicates the total number of pages that could be returned by repeated requests, based on what you specified for **EntriesPerPage**. The value in **Pagination.PageNumber** indicates the page number of the data returned in a call.

Find Items in a Category with GetCategoryListings

With GetCategoryListings, you can find items on eBay based on a category. You can filter results using many criteria, such as location, whether the item is listed in an eBay Store, and the listing type. The response is an array.

For example, you could use an URL similar to the following to retrieve items in category 20081:

```
http://rest.api.ebay.com/  
restapi?CallName=GetCategoryListings&RequestToken=UserToken&RequestUserId=UserName&CategoryID=20081&DetailLevel=ReturnAll&Version=491
```

You can use GetCategories to retrieve category information before you use GetCategoryListings (see “[Obtain a Category Hierarchy with GetCategories](#)” on page 18).

Parameters for GetCategoryListings are the same in the REST API as in the unified schema, except for the basic REST input parameters. For the REST-specific input parameters, see “[Basic REST Input Parameters](#)” on page 3. For more information about using GetCategoryListings, including descriptions of the call-specific input and output parameters, see the following document:

[GetCategoryListings](#)

General information about the unified schema format is available at the following location:

<http://developer.ebay.com/support/docs/>

After you have tested your program’s ability to handle output, you can use your program in production to transform output or you can enable server-side processing. Server-side processing uses your XSL files to format the output for you. For more information on server-side processing that uses your XSL files, see “[Transforming Data Returned](#)” on page 20.

Find Items with GetSearchResultsExpress

You can find items on eBay Express (“Express”) using the GetSearchResultsExpress call. Specifically, you can use GetSearchResultsExpress to do the following:

- Search for items on Express
- Search for eBay catalog products on Express
- Search and browse Express product types, aisles, and departments

For example, to use the **Query** parameter to search by keywords, you can specify an URL similar to the following:

```
http://rest.api.ebay.com/  
restapi?CallName=GetSearchResultsExpress&RequestToken=UserToken&RequestUserId=UserName&Que  
ry=plates&EntriesPerPage=3&Version=491
```

For additional samples for this call, see “[GetSearchResultsExpress Samples](#)” on page 14.

GetSearchResultsExpress retrieves brief details about active items that are listed on eBay Express. This call also retrieves eBay catalog products used to pre-fill data in these items, and a histogram that shows the distribution of items and catalog products across Express departments, aisles, and/or product types. Parameters such as `ItemDetails` control the fields to retrieve.

GetSearchResultsExpress is not available in the Sandbox.

Parameters for GetSearchResultsExpress are the same in the REST API as in the unified schema, except for the REST-specific input parameters. For the REST-specific input parameters, see “[Basic REST Input Parameters](#)” on page 3. For more information about using GetSearchResultsExpress, including descriptions of the call-specific input and output parameters, see the following document:

[GetSearchResultsExpress](#)

General information about the unified schema format is available at the following location:

<http://developer.ebay.com/support/docs/>

The following location contains additional information about using GetSearchResultsExpress in the unified schema, as well as links to related topics:

[http://developer.ebay.com/DevZone/XML/docs/WebHelp/
index.htm?context=eBay_XML_API&topic=ExpressSearch](http://developer.ebay.com/DevZone/XML/docs/WebHelp/index.htm?context=eBay_XML_API&topic=ExpressSearch)

After you have tested your program’s ability to handle output, you can use your program in production to transform output or you can enable server-side processing. Server-side processing uses your XSL files to format the output for you. For more information on server-side processing that uses your XSL files, see “[Transforming Data Returned](#)” on page 20.

GetSearchResultsExpress Samples

You can use the GetSearchResultsExpress call in many ways to obtain data from eBay Express.

GetSearchResultsExpress in the REST API has many similarities to GetSearchResultsExpress in the unified schema. A link to more information, which show more ways to use this call, is in the following topic. This unified schema topic contains detailed information about the call and can answer many questions:

http://developer.ebay.com/DevZone/XML/docs/WebHelp/index.htm?context=eBay_XML_API&topic=ExpressSearch

The following sample shows how to use the **Query** parameter to search by keywords:

<http://rest.api.ebay.com/restapi?CallName=GetSearchResultsExpress&RequestToken=UserToken&RequestUserId=UserName&Query=garmin+gps&EntriesPerPage=3&Version=491>

The following sample shows how to retrieve all the departments, aisles, and product types that are currently on Express. To make sure all the details are retrieved, `HistogramDetails` is set to `Fine`.

<http://rest.api.ebay.com/restapi?CallName=GetSearchResultsExpress&RequestToken=UserToken&RequestUserId=UserName&Version=491&DepartmentName=Express&HistogramDetails=Fine&HistogramSort=Alphabetical>

The following sample shows how to sort items in the GetSearchResultsExpress response by the lowest total cost, which includes the current fixed price plus default shipping costs. This request sets `ItemSort` to `LowestTotalCost` and it includes a postal code in `PostalCode` to enable Express to calculate shipping costs. This example also shows the effects of setting `HistogramDetails`, `ProductDetails`, and `ItemDetails` to `Fine`.

http://rest.api.ebay.com/restapi?CallName=GetSearchResultsExpress&RequestToken=UserToken&RequestUserId=UserName&Query=garmin+gps+receiver&EntriesPerPage=3&Version=491&DepartmentName=Consumer_Electronics&HistogramDetails=Fine&HistogramSort=Alphabetical&ProductDetails=Fine&ProductSort=LowestPrice&ItemDetails=Fine&ItemSort=LowestTotalCost&PostalCode=95125

The following sample shows how to search for a book with Pre-filled Item Information by using an ISBN as an external product ID. You can also use this approach to search for items by UPC. Please note that the ExternalProductID query only finds items that were listed with Pre-filled Item Information. That is, if the seller did not pre-fill an item's Item Specifics based on a stock catalog product, this query will not find the item.

<http://rest.api.ebay.com/restapi?CallName=GetSearchResultsExpress&RequestToken=UserToken&RequestUserId=UserName&EntriesPerPage=3&Version=491&ExternalProductIDType=ISBN&ExternalProductIDValue=0439425239&ProductDetails=Fine&ItemDetails=Fine&Condition=All>

Retrieve Product Information with GetProducts

You can use GetProducts to retrieve a variety of product information. GetProducts retrieves more buying guide details (as available) than GetSearchResults (described in “Find Items with GetSearchResults” on page 9).

GetProducts includes input parameters for retrieving the following types of product information:

- Stock catalog information about a particular DVD, digital camera, or a variety of other products
- Product titles, stock photos, and Item Specifics
- URLs to help you create links to additional catalog content on eBay
- Up to 200 active items that were listed with a product
- Up to 20 of the top reviews written by eBay members

For example, to use the **QueryKeywords** parameter to search for products that match "harry potter," you can specify an URL similar to the following:

```
http://rest.api.ebay.com/  
restapi?CallName=GetProducts&RequestToken=UserToken&RequestUserId=UserName&ProductSearch.Qu  
eryKeywords=Harry%Potter&ProductSearch.Pagination.EntriesPerPage=2&ProductSearch.Pagination.Pag  
eNumber=1&Version=505
```

The response includes two products. Each product returns a ProductReferenceID that can be used as input to GetProducts to retrieve reviews, items associated with the product, or both. Some types of products, like books, also return an ExternalProductID that can be used as an alternative to ProductReferenceID. Each product also return an ItemSpecifics node, which provides some standard information about the product. For example, book products usually return book-specific information such as the author. DVDs usually return movie-specific information such as the leading role.

You can use the ProductReferenceID from a product retrieved in the above example, as follows, to retrieve product reviews and to retrieve items listed with the product:

```
http://rest.api.ebay.com/  
restapi?CallName=GetProducts&RequestToken=UserToken&RequestUserId=UserName&ProductSearch.Pr  
oductReferenceID=44462740&IncludeItemArray=true&IncludeReviewDetails=true&Version=505
```

For samples of GetProducts that are used in the unified schema, please see the following location:

```
http://developer.ebay.com/DevZone/XML/docs/WebHelp/  
index.htm?context=eBay_XML_API&topic=GetProductsSamples
```

Parameters for GetProducts are the same in the REST API as in the unified schema, except for the REST-specific input parameters. For the REST-specific input parameters, see “Basic REST Input Parameters” on page 3. For more information about using GetProducts, including descriptions of the call-specific input and output parameters, see the following document:

[GetProducts](#)

General information about the unified schema format is available at the following location:

```
http://developer.ebay.com/support/docs/
```

The following location contains additional information about using GetProducts in the unified schema, as well as links to related topics:

http://developer.ebay.com/DevZone/XML/docs/WebHelp/index.htm?context=eBay_XML_API&topic=SearchingItemsWithQuery

After you have tested your program's ability to handle output, you can use your program in production to transform output or you can enable server-side processing. Server-side processing uses your XSL files to format the output for you. For more information on server-side processing that uses your XSL files, see ["Transforming Data Returned"](#) on page 20.

Obtain Item Data with GetItem

You can use GetItem to obtain detailed information about one item. GetItem takes an item ID as input.

For example, you could use an URL similar to the following to obtain detailed information about the item with the item ID of 123456789123:

<http://rest.api.ebay.com/restapi?CallName=GetItem&RequestToken=UserToken&RequestUserId=UserName&ItemID=123456789123&Version=491>

Optionally, you can specify other parameters, such as a detail level, to modify the data returned. For example, to use the **DetailLevel** parameter that is specifically for getting attribute information, you could specify an URL similar to the following:

<http://rest.api.ebay.com/restapi?CallName=GetItem&RequestToken=UserToken&RequestUserId=UserName&ItemID=123456789123&DetailLevel=ItemReturnAttributes&Version=491>

Parameters for GetItem are the same in the REST API as in the unified schema, except for the REST-specific input parameters. For the REST-specific input parameters, see ["Basic REST Input Parameters"](#) on page 3. For more information about using GetItem, including descriptions of the call-specific input and output parameters, see the following document:

[GetItem](#)

General information about the unified schema format is available at the following location:

<http://developer.ebay.com/support/docs/>

After you have tested your program's ability to handle output, you can use your program in production to transform output or you can enable server-side processing. Server-side processing uses your XSL files to format the output for you. For more information on server-side processing that uses your XSL files, see ["Transforming Data Returned"](#) on page 20.

Obtain Shipping Costs with GetItemShipping

You can use `GetItemShipping` to obtain estimated shipping cost information for an item. To use `GetItemShipping`, include the applicable item ID.

The data returned is similar to the data returned by the Shipping Calculator on the eBay site. Note that the shipping insurance cost can only be determined after the final price of the item is determined.

For example, using item ID 123456789123, you could specify destination information in a URL similar to the following for estimated shipping cost information:

```
http://rest.api.ebay.com/  
restapi?CallName=GetItemShipping&RequestToken=UserToken&RequestUserId=UserName&ItemID=123  
456789123&DestinationPostalCode=40404&DestinationCountryCode=US&Version=491
```

Parameters for `GetItemShipping` are the same in the REST API as in the unified schema, except for the basic REST input parameters. For the REST-specific input parameters, see “[Basic REST Input Parameters](#)” on page 3. For more information about using `GetItemShipping`, including descriptions of the call-specific input and output parameters, see the following document:

[GetItemShipping](#)

General information about the unified schema format is available at the following location:

```
http://developer.ebay.com/support/docs/
```

After you have tested your program’s ability to handle output, you can use your program in production to transform output or you can enable server-side processing. Server-side processing uses your XSL files to format the output for you. For more information on server-side processing that uses your XSL files, see “[Transforming Data Returned](#)” on page 20.

Obtain a Category Hierarchy with GetCategories

`GetCategories` retrieves the latest category hierarchy available on a site. You can retrieve all categories on the site, or one particular category and its subcategories.

You can use `GetCategories` to retrieve category information before you use `GetCategoryListings` (see “[Find Items in a Category with GetCategoryListings](#)” on page 13). For example, you can specify the ID of the highest-level category you want returned in order to return that category and its subcategories:

```
http://rest.api.ebay.com/  
restapi?CallName=GetCategories&RequestToken=UserToken&RequestUserId=UserName&CategoryParent  
=20081&DetailLevel=ReturnAll&Version=491
```

`GetCategories` can return a large amount of data in the response. To use this call, see the article “[Best Practices for GetCategories API call](#)” in the Developers Program Knowledge Base. Although the article applies to the call in the unified schema, you can convert the unified schema XML code in the article

by using an URL instead of XML and adding the necessary REST parameters (see “Basic REST Input Parameters” on page 3):

https://ebay.custhelp.com/cgi-bin/ebay.cfg/php/enduser/std_adp.php?p_faqid=597

For a category, GetCategories returns the category name, its unique ID (within the site you specify), and meta-data that helps you determine where the category fits in the category hierarchy.

Parameters for GetCategories are the same in the REST API as in the unified schema, except for the basic REST input parameters. For the REST-specific input parameters, see “Basic REST Input Parameters” on page 3. For more information about using GetCategories, including descriptions of the call-specific input and output parameters, see the following document:

[GetCategories](#)

General information about the unified schema format is available at the following location:

<http://developer.ebay.com/support/docs/>

Obtain Keywords with GetPopularKeywords

Using GetPopularKeywords, you can obtain a list of keywords that users have frequently specified when searching eBay. The list is regularly updated.

Optionally, you can specify a category ID for keywords relating to a specific category. Please note that you cannot specify the **CategoryID** parameter more than once per call.

To use the **CategoryID** parameter to obtain frequently-used keywords, you could specify an URL similar to the following:

[http://rest.api.ebay.com/
restapi?CallName=GetPopularKeywords&RequestToken=UserToken&RequestUserId=UserName&Category
ID=20081&Version=491](http://rest.api.ebay.com/restapi?CallName=GetPopularKeywords&RequestToken=UserToken&RequestUserId=UserName&CategoryID=20081&Version=491)

The following sample shows how to obtain frequently-used keywords for a category and its subcategories:

[http://rest.api.ebay.com/
restapi?CallName=GetPopularKeywords&RequestToken=UserToken&RequestUserId=UserName&Category
ID=58058&IncludeChildCategories=true&Version=491](http://rest.api.ebay.com/restapi?CallName=GetPopularKeywords&RequestToken=UserToken&RequestUserId=UserName&CategoryID=58058&IncludeChildCategories=true&Version=491)

Parameters for GetPopularKeywords are the same in the REST API as in the unified schema, except for the basic REST input parameters. For the REST-specific input parameters, see “Basic REST Input Parameters” on page 3. For more information about using GetPopularKeywords, including descriptions of the call-specific input and output parameters, see the following document:

[GetPopularKeywords](#)

General information about the unified schema format is available at the following location:

<http://developer.ebay.com/support/docs/>

After you have tested your program's ability to handle output, you can use your program in production to transform output or you can enable server-side processing. Server-side processing uses your XSL files to format the output for you. For more information on server-side processing that uses your XSL files, see ["Transforming Data Returned"](#) on page 20.

Transforming Data Returned

Using the XML-to-HTML Transform tool, you can enable processing that transforms the XML returned by the REST API. This server-side processing is done using XSL files that you upload. You access your uploaded XSL files using a link on the REST Developer Center page (<http://developer.ebay.com/developercenter/rest>), on the eBay Developers Program site.

You can use your uploaded XSL files to efficiently transform data returned into formats that meet your needs and the needs of your users. Server-side transformation of data returned in the REST API can remove the need for you to manage formatting and other transformations of data returned, thus providing you with enhanced flexibility and lower costs.

To enable server-side processing of your uploaded XSL files, you must have production keys and at least one production-certified application. For more information, see ["Using the REST API in Production"](#) on page 8. Server-side processing of uploaded XSL files is not available in the Sandbox environment.

Calls to the eBay REST API that utilize server-side processing are made using the following URL:

<http://www.ebayxslt.com/ws/web/apixslt?>

If you make a call to the eBay REST API that utilizes server-side processing, and omit the UnifiedInput parameter, the data returned is in unified schema format whether or not you add the Schema=1 setting. For information on the Schema=1 setting, see ["Basic REST Input Parameters"](#) on page 3.

This section of the REST API Guide contains the following subsections:

["Uploading and Using XSL Files"](#) on page 20

["Viewing, Renaming, and Deleting XSL Files"](#) on page 21

Uploading and Using XSL Files

After you have production keys and at least one production-certified application, you can upload XSL files to the XML-to-HTML Transform tool. The XML-to-HTML Transform tool is accessible from a link on the REST Developer Center page (<http://developer.ebay.com/developercenter/rest>).

You use the XML-to-HTML Transform tool by entering a production Developer Id (DevId) key. The DevId key is the first key of the DevId/AppId/CertId set of keys. It is an alphanumeric string. After you enter your DevId, you can upload the XSL files that eBay servers will use to transform data returned by the REST API. The eBay servers will use the Saxon 6.5.2 processor for transformations.

Each file you upload is associated with a production-certified application. A maximum of 20 XSL files can be associated with one production-certified application.

The smaller the size of an XSL file, the better will be its performance in transforming data returned. The maximum size for an uploaded XSL file is 2 MB, but a maximum size of 1 MB is recommended.

Step 1: Upload an XSL File

- 1 Enter your production Developer Id (DevId) in the XML-to-HTML Transform tool, accessible from the REST Developer Center page (<http://developer.ebay.com/developercenter/rest>).
- 2 If multiple applications are associated with your DevId, the applications are displayed. From the list of applications, select an application with which to associate your XSL file and click Next. If only one application is associated with your DevId, the REST XSLT Home page is displayed.
- 3 On the REST XSLT Home page, the name of your application is displayed. Click Upload New File.
- 4 On the Upload A New File page, specify a unique File Tag (such as FormatInHTML1) and the file path of an XSL file, and then click Upload. After the upload completes, the status of the upload, including any associated error messages, is displayed under the Upload button.
- 5 Click Go Back to return to the REST XSLT Home page. The File Tag of the file you uploaded is listed in the File Tag column of the table that displays information about your XSL files.

Step 2: Use an XSL File to Transform Data Returned

- 1 Test a production URL (without trying to utilize your XSL file) using the standard URL for the REST API. For more information, see “Basic REST Input Parameters” on page 3. Use the same values for `RequestToken` and `RequestUserId` that you will use in the call that includes the parameter for transforming data returned.
- 2 Complete your successfully tested production URL so it will result in transformed data returned.
 - 2a Replace the `http://rest.api.ebay.com/restapi?` portion of the URL with `http://www.ebayxslt.com/ws/web/apixslt?`
 - 2b Add `&filetag=FormatInHTML1` to the end of the URL, replacing `FormatInHTML1` with the File Tag of the file you uploaded in Step 1. For example, the URL for a `GetSearchResults` call might be similar to this:
<http://www.ebayxslt.com/ws/web/apixslt?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=toy%20boat&filetag=ResultsInHTML&Version=491&UnifiedInput=1>
- 3 Use the completed URL in your browser to make a call to the REST API.

Viewing, Renaming, and Deleting XSL Files

After you have uploaded XSL files, you can view them, rename their file tags, and delete them as necessary. (For information on uploading XSL files, see “Uploading and Using XSL Files” on page 20.)

- “View an XSL File” on page 22
- “Rename an XSL File Tag” on page 22
- “Delete an XSL File” on page 22

View an XSL File

To view an XSL file that you have uploaded:

- 1 Enter your production Developer Id (DevId) in the XML-to-HTML Transform tool, accessible from the REST Developer Center page (<http://developer.ebay.com/developercenter/rest>).
- 2 If multiple applications are associated with your DevId, the applications are displayed. From the list of applications, select the application with which your XSL file is associated and click Next. If only one application is associated with your DevId, the REST XSLT Home page is displayed.
- 3 On the REST XSLT Home page, the name of your application is displayed. Select an XSL file to view by clicking its File Tag. Note that the names of the files (that is, the file names associated with the File Tags) are not used to track XSL files. After an XSL file is uploaded, the file is tracked by a File Tag, not by a file name.
- 4 On the View XSL file page, you can view the file but you cannot modify it. If you want to change the file, copy its text to a text file, save the file, and see “[Uploading and Using XSL Files](#)” on page 20 for information about uploading the file.
- 5 When you are finished viewing the file, click Go Back to return to the REST XSLT Home page.

Rename an XSL File Tag

To rename the File Tag of an XSL file that you have uploaded:

- 1 Enter your production Developer Id (DevId) in the XML-to-HTML Transform tool, accessible from the REST Developer Center page (<http://developer.ebay.com/developercenter/rest>).
- 2 If multiple applications are associated with your DevId, the applications are displayed. From the list of applications, select the application with which your XSL file is associated and click Next. If only one application is associated with your DevId, the REST XSLT Home page is displayed.
- 3 On the REST XSLT Home page, the name of your application is displayed. Select an XSL File Tag to change by clicking a current File Tag.
- 4 On the View XSL file page, click Rename.
- 5 On the Rename File page, enter a new File Tag and click Rename.
- 6 On the View XSL file page, the new File Tag is displayed above the Rename button.

Delete an XSL File

To delete an XSL file that you have uploaded:

- 1 Enter your production Developer Id (DevId) in the XML-to-HTML Transform tool, accessible from the REST Developer Center page (<http://developer.ebay.com/developercenter/rest>).
- 2 If multiple applications are associated with your DevId, the applications are displayed. From the list of applications, select an application with which your XSL file is associated and click Next. If only one application is associated with your DevId, the REST XSLT Home page is displayed.
- 3 On the REST XSLT Home page, the name of your application is displayed. Select an XSL file to delete using the associated checkbox. You can select multiple files to delete at once.
- 4 Click Delete. The status of the deletion is displayed under the Upload New File button.

Legacy REST GetSearchResults Input

Formerly, REST input parameters for GetSearchResults were similar to the legacy XML API input parameters for GetSearchResults.

Now, all applications must be migrated from legacy input parameters to unified schema input parameters. The unified schema is standardized and uses objects for data exchange.

This section of the REST API Guide may be applicable until you migrate all applications from legacy input parameters to unified schema input parameters.

This section contains the following subsections:

[“Legacy Samples for GetSearchResults”](#) on page 23

[“Legacy Affiliate Parameters for GetSearchResults”](#) on page 23

[“Legacy Input Parameters for GetSearchResults”](#) on page 24

Legacy Samples for GetSearchResults

This section contains samples that use legacy GetSearchResults input parameters. All applications must be migrated from legacy input parameters to unified schema input parameters. Please see the following section of this guide for the unified schema GetSearchResults input parameters: [“GetSearchResults Input Parameters”](#) on page 10. Also see the section [“Root URLs, Containers, and Repeating Parameters”](#) on page 3.

This legacy input sample shows use of a **Query** parameter to search eBay. The value of **Query** is "toy boat" with the space character HTTP-encoded as %20:

```
http://rest.api.ebay.com/  
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=toy  
%20boat&Schema=1&CompatibilityLevel=475
```

This legacy input sample shows use of a category ID in the request:

```
http://rest.api.ebay.com/  
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Category=6  
17&Query=star+trek&Schema=1&CompatibilityLevel=475
```

This legacy input sample shows a search for listing titles and descriptions that contain "fuzzy slippers"; a maximum of 5 results is returned:

```
http://rest.api.ebay.com/  
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=fuzz  
y%20slippers&MaxResults=5&SearchInDescription=1&Schema=1&CompatibilityLevel=475
```

Legacy Affiliate Parameters for GetSearchResults

This sample shows use of affiliate parameters in GetSearchResults using legacy input parameters:

```
http://rest.api.ebay.com/  
restapi?CallName=GetSearchResults&RequestToken=UserToken&RequestUserId=UserName&Query=tape
```

`&TrackingProvider=MyTrackingProvider&TrackingId=MyTrackingId&CJSID=MyCJSID&Schema=1&CompatibilityLevel=491`

All applications must be migrated from legacy input parameters to unified schema input parameters. Please see the following section of this guide for the unified schema GetSearchResults input parameters: “[GetSearchResults Input Parameters](#)” on page 10. The following table describes the legacy GetSearchResults affiliate parameters.

Table 1-4 Legacy GetSearchResults Parameters for Affiliate Tracking

Parameter	Required?	Data Type	Purpose
TrackingProvider	N	Integer	<p>Specifies the third party, e.g. Commission Junction, who is your tracking partner (tracking provider).</p> <p>Depending on your tracking partner, specify one of the following values. Not all partners are valid for all sites.</p> <ul style="list-style-type: none"> 1 = Commission Junction 2 = Be Free 3 = Affilinet 4 = TradeDoubler 5 = Mediaplex 6 = DoubleClick 7 = Allyes 8 = BJMT
TrackingId	N	String	<p>Specifies an ID to identify you to your tracking partner (tracking provider).</p> <p>The value you specify is obtained from your tracking partner. For example, in the case of Commission Junction, this parameter is the PID given to you by Commission Junction.</p>
CJSID	N	String	<p>Optional value that specifies the Commission Junction Shopper ID (SID). This parameter is applicable if you used 1 (Commission Junction) for the TrackingProvider parameter.</p> <p>CJSID is a user-defined ID number for identifying specific search-result placement of Commission Junction affiliates. You can specify up to 30 characters (plus HTML-encoded spaces, commas, periods, and the “@” character), but you cannot specify zero or non-alpha-numeric characters.</p>

Legacy Input Parameters for GetSearchResults

All applications must be migrated from legacy input parameters to unified schema input parameters. Please see the following section of this guide for the unified schema GetSearchResults input parameters: “[GetSearchResults Input Parameters](#)” on page 10. Formerly, REST input parameters for GetSearchResults were similar to the Legacy XML API input parameters for GetSearchResults. Now, you can specify unified schema parameters for GetSearchResults. The unified schema is standardized and uses objects for data exchange.

Until you migrate all REST applications from legacy input parameters to unified schema input parameters, this section of the REST API Guide may be applicable to GetSearchResults calls because this section contains information about legacy GetSearchResults input.

The following table describes major legacy input parameters for GetSearchResults in the REST API.

Table 1-5 Legacy GetSearchResults Input Parameters (See “GetSearchResults Input Parameters” on page 10)

Parameter	Required?	Data Type	Purpose
Category	Y/N	Integer	Numeric ID of the category to be searched. Defaults to all categories if zero (0) is specified or the tag is not present. If the specified category ID does not match an existing category for the site, an invalid-category error message is returned.
CompatibilityLevel	Y	String	Specifies the API version you are using for the call. If data returned is different from expected, test your call using a higher version, such as 491.
CharityListing	N	Boolean	If 1 (true), limits the results to Giving Works listings (and no other listings). If 0 (false), has no effect on the search results (i.e., the behavior is the same as not sending CharityListing at all). Not applicable (returns 0 items) for eBay Motors, international sites, and Real Estate and Tickets listings.
Currency	N	Integer	Numeric ID for a currency type. Limits the search to just those items listed using a specified currency. Not applicable to eBay Motors searches.
DetailLevel	N	Integer	Controls the amount of data returned.
EndTimeFrom	N	Time	Specifies the beginning of a time range for when items will end.
EndTimeTo	N	Time	Specifies the end of a time range for when items will end.
ErrorLanguage	N	Integer	Corresponds to a Site ID. If specified, returns any error messages in the language associated with the Site ID.
ExcludeSellers	N	String	Excludes the specified sellers from the search. Use a seller's User ID. Separate multiple sellers with a comma. A maximum of 100 sellers can be listed. Cannot be used with IncludeSellers and PayPal parameters. StoreSearch overrides ExcludeSellers.
ExpandSearch	N	Boolean	Expands search results when a small set is returned.
ExternalProductIdentifier	N	Meta Tag	Do not use the Query parameter with this parameter. ExternalProductIdentifier only retrieves items that were listed with the specified ISBN or UPC. Only applicable for items with Pre-filled Item Information in media categories such as Books and Music. In the REST API, you can use XPATH for parameters that are containers or have attributes (such as the ExternalProductIdentifier parameter). In the case of ExternalProductIdentifier, id and type are attributes because ExternalProductIdentifier is defined as <ExternalProductIdentifier id="value" type="value" />. An example of using XPATH is the following search for an ISBN of 0811645169: CallName=GetSearchResults&ExternalProductIdentifier@type=ISBN&ExternalProductIdentifier@id=0811645169.
GranularityLevel	N	String	Reduces the number of tags returned for items.
HighestPrice	N	Money	The highest price in the Current Price field that any item returned by the Search results should have.

Table 1-5 Legacy GetSearchResults Input Parameters (See “GetSearchResults Input Parameters” on page 10)

Parameter	Required?	Data Type	Purpose
<code>IncludeCondition</code>	N	Boolean	When passed with a value of 1 (true), each item in the result set can also include the item condition attribute (e.g., New or Used), if the seller specified it. In this case, an item will only include the condition attribute if the item's seller filled in the listing's Item Condition (in the Item Specifics section of the listing). (That is, the condition attribute is not returned if the seller only put the word "New" or "Used" in the listing's title but did not specify the condition in the listing's Item Specifics.) If 0 or not specified, the item condition is not returned. Note that this field does not control whether to retrieve only new or used items. In some categories, you can use a product finder search (<code>SearchRequest</code>) to only retrieve new or used items.
<code>IncludeFeedback</code>	N	Boolean	When passed with a value of 1 (true), each item in the result set also includes information about the seller's feedback (the positive feedback percentage, whether or not the seller has chosen to keep their feedback private, and the feedback score).
<code>IncludeGetItFast</code>	N	Boolean	When passed with a value of 1 (true), restricts listings returned to Get It Fast listings.
<code>IncludeSellers</code>	Y/N	String	Restricts search to the sellers specified. Specify the seller's user ID. Separate multiple sellers with a comma. A maximum of 100 sellers can be listed. Cannot be used with <code>ExcludeSellers</code> and <code>PayPal</code> parameters. <code>StoreSearch</code> overrides <code>IncludeSellers</code> .
<code>IsLot</code>	N	Boolean	When passed with a value of 1 (true), restricts listings returned to only those for which <code>LotSize</code> is 2 or greater.
<code>ItemTypeFilter</code>	N	Integer	Filters items returned by the type of listing. Valid values are: 1 = Auction items only 2 = Fixed price items (both Fixed Price and Buy It Now) 3 = All items
<code>LowestPrice</code>	N	Money	The lowest price in the <code>Current Price</code> field that any item returned by the Search results should have.
<code>MaxDistance</code>	N	Integer	Maximum distance to use for sorting by distance. This is used in proximity searches.
<code>MaxResults</code>	N	Integer	Maximum number of items matching search criteria to return. You can specify up to 400 items per call. The valid range for this parameter is 1 to 400 and the default is 100. (Valid range for eBay Motors searches: 1 to 30.) Use <code>Skip</code> to specify the number of items to skip when building the result set. Use <code>MaxResults</code> and <code>Skip</code> together to paginate your results set.
<code>Modifier</code>	N	String	Used with the <code>ModifierCode</code> parameter. The <code>SiteIdFilterType</code> parameter is a more current filter than <code>Modifier</code> .
<code>ModifierCode</code>	N	String	Used with the <code>Modifier</code> parameter. The <code>SiteIdFilterType</code> parameter is a more current filter than <code>Modifier</code> .
<code>ModTimeFrom</code>	N	Time	Filter that limits the response to active items whose status has changed since the specified time.

Table 1-5 Legacy GetSearchResults Input Parameters (See “GetSearchResults Input Parameters” on page 10)

Parameter	Required?	Data Type	Purpose
MotorsGermanySearchable	N	Boolean	Filter that limits the response according to whether the item is eligible to be listed in search results on the mobile.de site.
NowAndNew	N	Boolean	Filter that limits the response to items listed with Now & New.
Order	N	String	Specifies the sort order of the returned list of items. Default is MetaEndSort. Not all values can be used with eBay Motors searches.
PayPal	N	Boolean	When passed with a value of 1 (true), restricts items returned to only those where PayPal is accepted as a buyer payment option. Default is 0 (false). Cannot be used with <code>IncludeSellers</code> and <code>ExcludeSellers</code> arguments. Applicable when searching sites that support PayPal (ignored for sites that do not support PayPal)
PostalCode	N/Y	String	The zip or postal code from which to compute the distance. This is required when constraining searches with proximity.
ProductId	Y/N	String	Exclusive filter to retrieve items that were listed with the specified eBay catalog product. If you specify a product ID that is no longer in use, eBay will use the product's current ID (if available) instead (with no warning). You must specify <code>ProductId</code> , <code>Query</code> , <code>ExternalProductIdIdentifier</code> , <code>IncludeSellers</code> , or <code>Category</code> in the request. <code>ProductId</code> cannot be used in combination with <code>Query</code> , <code>ExternalProductIdIdentifier</code> , or <code>Category</code> .
Query	Y/N	String	Specifies the search string. The search string consists of one or more keywords to search for in the listing title and/or description. Use characters in the UTF-8 character set. The words "and" and "or" are treated like any other word. Only use "and", "or", or "the" if you are searching for listings containing these words. You can use AND or OR logic by including certain modifiers. Wildcards (e.g., *) are also supported. Be careful when using spaces before or after modifiers and wildcards (+, -, or *).
Quantity	N	Integer	The total number of objects desired. If <code>IsLot</code> is also specified, then <code>Quantity</code> is the number of lots times the number of objects in each lot. Use <code>QuantityOp</code> to specify that you are seeking quantities greater than, equal to, or less than <code>Quantity</code> .
QuantityOp	N	String	An operator that specifies the relationship of the desired quantity to the integer value in <code>Quantity</code> . Valid values are <code>lt (<)</code> , <code>le (<=)</code> , <code>eq (=)</code> , <code>gt (>)</code> , and <code>ge (>=)</code> .
Region	N	Integer	Numeric ID for a region. Limits the search to items listed in a specified region.
RestScript	N	Boolean	If true (1), specifies to surround the entire search results with <code>document.write (“”)</code> . This is useful in certain situations where you want to use scripting on the result set. Defaults to false (0).
Schema	Y	Boolean	Specify <code>Schema=1</code> to return data in the unified schema format. (If omitted, or if you set it to 0, the results are returned in the Legacy XML API format, and output in that format is not supported as of June 1, 2006.)

Table 1-5 Legacy GetSearchResults Input Parameters (See “GetSearchResults Input Parameters” on page 10)

Parameter	Required?	Data Type	Purpose
SearchInDescription	N	Boolean	If true (1), specifies to search in item titles, subtitles, and descriptions for the string specified in Query. Defaults to false (0) (title search only)
SearchType	N	Integer	Restricts items returned to, selectively, only gallery items or gallery and non-gallery items. Valid values are: 0 = Search for gallery and non-gallery items. 1 = Search for gallery items only and display a preview (thumbnail) image of gallery picture.
SellerBusinessType	N	Integer	Limits the listings to those of a specified type of business. Possible values: 0 = Undefined 1 = Private 2 = Commercial
SiteId	N	Integer	Specifies the eBay site to be searched and from which to retrieve items meeting the search criteria. If not specified, then the default (US) eBay site is searched.
SiteIdFilterType	N	Integer	Allows filtering of category listings by SiteId.
Skip	N	Integer	Specifies the number of items to skip over (counting from the first) in paginating results.
StoreID	N	Integer	Restricts the search to the specified Store. Only applicable in the Sandbox. You must specify a value in StoreID in the Sandbox if StoreSearch = 0, 1, or 2.
StoreName	N	String	Restricts the search to the specified Store. You must specify a value in StoreName if StoreSearch = 0, 1, or 2. The StoreName parameter is case sensitive. The maximum length is 46.
StoreSearch	N	Integer	Restricts the search to items that are listed in eBay Stores (Stores Inventory items and/or fixed-price and auction items that also appear in a seller's eBay Store). Use the StoreSearch value to specify which store items to return. Using StoreSearch forces SearchInDescription = true. 0 = Within a single store for all items (use with StoreName). 1 = Within a single store for auction items (use with StoreName). 2 = Within a single store for Buy It Now items (use with StoreName). 3 = Across all stores for Buy It Now items.
TotalOnly	N	Boolean	Filter to return only the total quantity of matching items (Count, HasMoreItems, PageNumber, GrandTotal), without returning the item data or buying guide data.

Chapter 2

Additional Resources

In each REST call, you provide the input parameters in the request URL; the parameters are part of the HTTP request. The response contains the requested data in XML format. This contrasts with other Web service architectures, such as those that use SOAP, in which clients submit data to the service as an XML document.

Unlike other eBay APIs, the REST API uses HTTP (instead of HTTPS). This is because unlike other eBay APIs, the REST API is used to access only publicly-accessible data.

REST Information Sources

The following contain additional information on the REST architecture:

REST Discussion Mailing List

<http://groups.yahoo.com/group/rest-discuss/>

Roy Thomas Fielding's PhD Dissertation

<http://www1.ics.uci.edu/~fielding/pubs/dissertation/top>

"Building Web Services the REST Way"

<http://www.xfront.com/REST-Web-Services.html>

"REST Architectural Style" wiki page

<http://c2.com/cgi/wiki?RestArchitecturalStyle>

"RESTWiki"

<http://rest.blueoxen.net/cgi-bin/wiki.pl?FrontPage>

eBay API Documentation

The eBay API documentation is available on the eBay Developers Program Web site.

eBay Developers Program Web site

<http://developer.ebay.com>

Index

E

- eBay Affiliate Program** 5
- eBay REST API**
 - required parameters 3
 - URL for the 3
 - URL for the Sandbox 3
 - URL for using transformations 20
 - using in a web page 14, 17
 - using in an application 14, 17

G

- GetSearchResults output fields** 12
- GetSearchResults samples** 10

H

- HTTP protocol** 29

P

- production keys** 8

R

- REST**
 - additional information on 29
 - contrasted with SOAP web services 29
- REST token**
 - how to get a 8

S

- Sandbox**
 - REST API support 8
- search query modifiers** 12
- search results**
 - paginating 13

T

- tracking**
 - enabling in the search results 5
- transforming data returned** 20

U

- URL**
 - size limitation 3

