

BigCommerce XML API

Table of Contents

Introduction	2
Accessing the API	2
Requesting Data	2
Sample XML Request.....	2
Response Format	3
Sample XML Response (Success)	3
Sample XML Response (Error)	3
API Authentication	4
Accessing Orders	4
Searching for Orders.....	4
Sample XML Request.....	4
Sample XML Response	5
Search Options	6
Retrieving Order Details	7
Sample XML Request.....	7
Sample XML Response	7
Response Data.....	7
Accessing Products	8
Searching for Products.....	8
Sample XML Request.....	8
Sample XML Response	8
Search Options	9
Retrieving Product Details	10
Sample XML Request.....	10
Sample XML Response	11
Response Data.....	11
Accessing Customers.....	12
Searching for Customers	12
Sample XML Request.....	12
Sample XML Response	12
Search Options	13
Sample Request – PHP	14

Introduction

BigCommerce contains a simple data-out API that allows remote retrieval of customers, orders and products via a simple XML web service. The API allows requests to be made to search for the above data, or retrieve a particular entry from the database for one of the supported data types.

Accessing the API

Requesting Data

- Requests are made to an API end-point, available at **`http://your.store.com/xml.php`**, with `your.store.com` being the domain name/full address to your store.
- The request should be submitted as a standard HTTP POST operation, with the post data containing the request XML.
- Requests made should be structured of valid XML, and an XML response will always be returned.
- Each API request should contain a valid **username** and **usertoken** XML node, as described in *API Authentication* below.
- The API request must contain a valid **requesttype** (see the appropriate section below) and a valid **requestmethod** (also in each section below) with the action to perform.
- The data for the request (search parameters etc) should be supplied within a **details** XML node.

Sample XML Request

```
<xmlrequest>
  <username>admin</username>
  <usertoken>e30bfbbf36db9aff34e660a1cb322b23</usertoken>
  <requesttype>orders</requesttype>
  <requestmethod>GetOrders</requestmethod>
  <details>
    <start>1</start>
    <searchQuery>@bigcommerce.com</searchQuery>
  </details>
</xmlrequest>
```

Response Format

- An XML response from the API will always be returned. The root XML node of the API response is **response**.
- The XML response will contain a status code (available under the **status**) node of either **SUCCESS** or **FAILED**. The requesting application should determine the appropriate method to handle the response based on this status.
- In the case of a successful response, a **data** node will be present containing the returned XML from the API request (search results, etc)
- For a failed response, an **errormessage** node will be populated with the reason for why the request failed.

Sample XML Response (Success)

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
  <status>SUCCESS</status>
  <data>
    <orderid>1</orderid>
    <ordcustomername>Joe Bloggs</ordcustomername>
    <items>
      <item>
        <ordprodname>Example Product</ordprodname>
      </item>
      <item>
        <ordprodname>Another Example
Product</ordprodname>
      </item>
    </items>
  </data>
</response>
```

Sample XML Response (Error)

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
  <status>FAILED</status>
  <errormessage>The login details you supplied were
invalid.</errormessage>
</response>
```

API Authentication

Access to the BigCommerce XML API is granted on an authenticated user basis. That is, a username and a unique token need to be supplied with each request for a valid store administrator with API access.

To enable the BigCommerce XML API for a particular user, follow the steps below:

1. Login to the BigCommerce Control Panel
2. Click **User Accounts** in the top navigation menu to be taken to the listing of store administrators.
3. Click **Edit** next to the user for which API access should be granted.
4. Tick **Yes, allow this user to use the XML API** to enable API access.
5. Take note of the **XML Path** and **XML Token** values as these will be used to access the API and are unique to this user.
6. Click **Save** to save the API access settings for the user.

The authentication details are supplied within the **username** and **usertoken** nodes of the request XML.

Accessing Orders

All order operations should be performed using the **requesttype** value of **orders**. Supported actions include searching for orders or retrieving the details of a particular order.

Searching for Orders

To search for orders, the **requestmethod** should be supplied as **GetOrders**. Any search fields/options should be supplied in the **data** node of the XML request. A list of available search options can be found in *Search Options* below.

Sample XML Request

The XML request below will match all orders placed with an email address containing @bigcommerce.com. The results will start from the 5th matched record (**<start>5</start>**) and the default amount of products will be returned (20).

```
<xmlrequest>
  <username>admin</username>
  <usertoken>e30bfbbf36db9aff34e660a1cb322b23</usertoken>
  <requesttype>orders</requesttype>
```

```

<requestmethod>GetOrders</requestmethod>
<details>
  <start>5</start>
  <searchQuery>@bigcommerce.com</searchQuery>
</details>
</xmlrequest>

```

Sample XML Response

The response below corresponds with the above sample XML request. The orders matching the specified criteria are returned.

- The **start** node references the offset from the beginning of the results that these results are for.
- The **end** node references the end of the result set.
- The **numResults** node contains the total number of results that the search matched.
- The **results** node contains the matching orders based on the search criteria. Each result is encapsulated in an **item** tag.

```

<response>
  <status>SUCCESS</status>
  <data>
    <start>5</start>
    <end>20</end>
    <numResults>40</numResults>
    <results>
      <item>
        <orderid>3</orderid>
        <ordcustid>1</ordcustid>
        <orddate>1245289425</orddate>
        ...
      </item>
      <item>
        <orderid>4</orderid>
        <ordcustid>1</ordcustid>
        <orddate>1245289425</orddate>
        ...
      </item>
      ...
    </results>
  </data>
</response>

```

Search Options

Below is a listing of all of the available search options (nested under the <details> tag) for searching for orders placed within the store.

- **couponCode** – All orders with the specified coupon code.
- **newMessages** – Set to '1' to return orders with new order messages.
- **orderId** – Search for orders with a particular order ID.
- **customerId** – Search all orders placed by a particular customer (based on customer ID)
- **searchQuery** – Search for orders matching the specified criteria. Will search order ID (exact match), tracking no (exact match), payment provider (exact match), customer name, billing name, shipping name, customer email address, billing address or shipping address.
- **orderStatus** – Search for orders of a particular status (based on status ID)
- **paymentMethod** – Search for orders paid with a particular payment method.
- **shippingMethod** – Search for orders with a particular shipping method.
- **orderFrom** – Search for orders with an order ID greater than or equal to the specified number.
- **orderTo** – Search for orders with an order ID less than or equal to the specified number.
- **totalFrom** – Search orders with an order total greater than or equal to the specified amount.
- **totalTo** – Search for orders with an order total less than or equal to the specified amount.
- **dateRange** – Search for orders placed in the specified time period (acceptable values are today, yesterday, day, week, month, this_month, this_year)
- **fromDate** – Search for orders placed on or after a particular date (date must be in mm/dd/yyyy format)
- **toDate** – Search for orders placed up until a particular date (date must be in mm/dd/yyyy format)
- **productId** – Search for orders containing a particular product based on product ID.
- **productName** – Search for orders containing a particular product based on the product name (like match)

Retrieving Order Details

To retrieve all of the details regarding a particular order, the **requestmethod** should be supplied as **GetOrder**. The ID of the order to fetch the details for should be supplied within the **orderId** node inside the **data** node.

The returned response will include all of the order details, as outlined in *Response Data* below.

Sample XML Request

The XML request below will fetch the details regarding order #1 from the store.

```
<xmlrequest>
  <username>admin</username>
  <usertoken>e30bfbbf36db9aff34e660a1cb322b23</usertoken>
  <requesttype>orders</requesttype>
  <requestmethod>GetOrder</requestmethod>
  <details>
    <orderId>1</orderId>
  </details>
</xmlrequest>
```

Sample XML Response

The XML response below contains a successful response for fetching the details of order #1 from BigCommerce.

```
<?xml version="1.0" encoding="UTF-8" ?>
<response>
  <status>SUCCESS</status>
  <data>
    <orderid>1</orderid>
    <ordcustid>1</ordcustid>
    <orddate>1244956454</orddate>
    <ordsubtotal>29.00</ordsubtotal>
    <ordtaxtotal>0.00</ordtaxtotal>
    ...
  </data>
</response>
```

Response Data

The following items are included in the XML response:

- All of the order details, as stored in the **orders** table of BigCommerce.

- An **items** node containing a series of nested **item** tags containing each of the items in the order, as stored in the **order_products** table.
- If one or more items have been configured by the customer during checkout (configurable products), all of the configurable fields will be included in a **configurablefields** node. Each configurable field is enclosed in an **item** tag, and can be related back to the ordered product by the **ordprodid** node.

Accessing Products

All product operations should be performed using the **requesttype** value of **products**. Supported actions include searching for products or retrieving the details of a particular product.

Searching for Products

To search for products, the **requestmethod** should be supplied as **GetProducts**. Any search fields/options should be supplied in the **data** node of the XML request. A list of available search options can be found in *Search Options* below.

Sample XML Request

The XML request below will match all products, beginning from the first record (<start>0</start>) where the product search keywords (name, description etc) contains **ipod** and the price is less than **\$1,000.00**. The first 20 matching products will be returned.

```
<xmlrequest>
  <username>admin</username>
  <usertoken>e30bfbbf36db9aff34e660a1cb322b23</usertoken>
  <requesttype>products</requesttype>
  <requestmethod>GetProduct</requestmethod>
  <details>
    <start>0</start>
    <searchQuery>ipod</searchQuery>
    <priceTo>1000</priceTo>
  </details>
</xmlrequest>
```

Sample XML Response

The response below corresponds with the above sample XML request. The products matching the specified criteria are returned.

- The **start** node references the offset from the beginning of the results that these results are for.
- The **end** node references the end of the result set.
- The **numResults** node contains the total number of results that the search matched.
- The **results** node contains the matching products based on the search criteria. Each result is encapsulated in an **item** tag.

```

<response>
  <status>SUCCESS</status>
  <data>
    <start>0</start>
    <end>6</end>
    <numResults>6</numResults>
    <results>
      <item>
        <productid>1</productid>
        <prodname>iPod Nano</prodname>
        <prodprice>169.00</prodprice>
        ...
      </item>
      <item>
        <productid>2</productid>
        <prodname>iPod Shuffle</prodname>
        <prodprice>57.00</prodprice>
        ...
      </item>
      ...
    </results>
  </data>
</response>

```

Search Options

Below is a listing of all of the available search options (nested under the <details> tag) for searching for products within the store.

- **productId** – Search for products matching a particular product ID.
- **searchQuery** – Search for products matching the specified criteria. Will search product ID (exact match), product name or product code (exact match)
- **letter** – Search for products with a name beginning with the specified letter (specify 0-9 for products beginning with a letter other than the A-Z range)

- **soldFrom** – Search for products that have sold an amount greater than or equal to the specified number.
- **soldTo** – Search for products that have sold an amount less than or equal to the specified number.
- **priceFrom** – Search for products that have a price greater than or equal to the specified amount.
- **priceTo** – Search for the products that have a price less than or equal to the specified amount.
- **inventoryFrom** – Search for products with a stock level greater than or equal to the specified number.
- **inventoryTo** – Search for products with a stock level less than or equal to the specified number.
- **inventoryLow** – Supply as '1' to search for products that have reached their low inventory level.
- **brand** – Search for products of a particular brand, based on the supplied brand ID.
- **visibility** - Search for products with a particular visibility setting. Supply as '1' to search for visible products only, or '0' to search for products not marked as visible.
- **featured** – Search for products with a particular featured setting. Supply as '1' to search for featured products only, or '0' to search for products not marked as featured.
- **freeShipping** – Search for products with a particular free shipping status. Supply as '1' to search for products with free shipping only, or '0' to search for products without free shipping.

Retrieving Product Details

To retrieve all of the details regarding a particular product, the **requestmethod** should be supplied as **GetProduct**. The ID of the product to fetch the details for should be supplied within the **productId** node inside the **data** node.

The returned response will include all of the product details, as outlined in *Response Data* below.

Sample XML Request

The XML request below will fetch the details regarding product #1 from the store.

```
<xmlrequest>
  <username>admin</username>
  <usertoken>e30bfbfb36db9aff34e660a1cb322b23</usertoken>
```

```

<requesttype>products</requesttype>
<requestmethod>GetProduct</requestmethod>
<details>
  <productId>1</productId>
</details>
</xmlrequest>

```

Sample XML Response

The XML response below contains a successful response for fetching the details of product #1 from BigCommerce.

```

<?xml version="1.0" encoding="UTF-8" ?>
<response>
  <status>SUCCESS</status>
  <data>
    <productid>1</productid>
    <prodname>Example Product</prodname>
    <prodcode>SKU</prodcode>
    <imagefile>http://...</imagefile>
    <images>
      ...
    </images>
  </data>
</response>

```

Response Data

The following items are included in the XML response:

- The product details, as stored in the **products** table of BigCommerce.
- The product thumbnail image, as the **imagefile** node.
- The product brand name, as the **prodbrandname** node.
- An **images** node containing a series of nested **item** tags containing each of the images for the product, as stored in the **product_images** table.
- A **categories** node containing a series of nested **item** tags containing each of the categories the product is in.
- A **variations** node containing a series of nested **items** tags containing each of the variations (if any) that the product has associated.

Accessing Customers

All customer operations should be performed using the **requesttype** value of **customers**. Supported actions include searching for customers or retrieving the details of a particular customer.

Searching for Customers

To search for products, the **requestmethod** should be supplied as **GetCustomers**. Any search fields/options should be supplied in the **data** node of the XML request. A list of available search options can be found in *Search Options* below.

Sample XML Request

The XML request below will match all customers, beginning from the first record (**<start>0</start>**) where the customer search keywords (surname etc) contains **Bloggs**. The first 20 matching customers will be returned.

```
<xmlrequest>
  <username>admin</username>
  <usertoken>e30bfbbf36db9aff34e660a1cb322b23</usertoken>
  <requesttype>customers</requesttype>
  <requestmethod>GetCustomers</requestmethod>
  <details>
    <start>0</start>
    <searchQuery>Bloggs</searchQuery>
  </details>
</xmlrequest>
```

Sample XML Response

The response below corresponds with the above sample XML request. The customers matching the specified criteria are returned.

- The **start** node references the offset from the beginning of the results that these results are for.
- The **end** node references the end of the result set.
- The **numResults** node contains the total number of results that the search matched.
- The **results** node contains the matching customers based on the search criteria. Each result is encapsulated in an **item** tag.

```
<reponse>
  <status>SUCCESS</status>
```

```

<data>
  <start>0</start>
  <end>20</end>
  <numResults>100</numResults>
  <results>
    <item>
      <customerid>1</customerid>
      <custfirstname>Joe</custfirstname>
      <custlastname>Bloggs</custlastname>

      <custconemail>email@example.com</custconemail>
      ...
    </item>
    <item>
      <customerid>2</customerid>
      <custfirstname>Jane</custfirstname>
      <custlastname>Bloggs</custlastname>

      <custconemail>jane@example.com</custconemail>
      ...
    </item>
    ...
  </results>
</data>
</response>

```

Search Options

Below is a listing of all of the available search options (nested under the <details> tag) for searching for products within the store.

- **searchQuery** – Search for customers matching the specified criteria. Fields searched are customer ID (exact match), customer first name, customer last name, customer email address, customer full name or customer company.
- **letter** – Search for customers with a surname beginning with the specified letter (specify 0-9 for customers surnames beginning with a letter other than the A-Z range)
- **phone** – Search for customers with a phone number containing the specified string.
- **idFrom** – Search for customers with a customer ID greater than or equal to the supplied number.
- **idTo** – Search for customers with a customer ID less than or equal to the supplied number.

- **storeCreditFrom** – Search for customers having an amount of store credit greater than or equal to the supplied amount.
- **storeCreditTo** – Search for customers having an amount of store credit less than or equal to the supplied amount.
- **dateRange** – Search for customers who registered in the specified time period (acceptable values are today, yesterday, day, week, month, this_month, this_year)
- **fromDate** – Search for customers who registered on or after a particular date (date must be in mm/dd/yyyy format)
- **toDate** – Search for customers who registered up until a particular date (date must be in mm/dd/yyyy format)custGroupId -
- **country** – Search for customers from a particular country.
- **state** – Search for customers from a particular state (must include matching country)

Sample Request – PHP

```
<?php
$storeURL = 'http://www.store.com/xml.php';
$storeUser = 'admin';
$storeToken = 'e30bfbbf36db9aff34e660a1cb322b23';

$xml = '<xmlrequest>
    <username>'.$storeUser.'</username>
    <usertoken>'.$storeToken.'</usertoken>
    <requesttype>orders</requesttype>
    <requestmethod>GetOrder</requestmethod>
    <details>
        <orderId>1</orderId>
    </details>
</xmlrequest>';

$ch = curl_init($storeURL);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_TIMEOUT, 60);
curl_setopt($ch, CURLOPT_POSTFIELDS, $xml);

echo '<pre>';
echo htmlspecialchars(curl_exec($ch));
echo '</pre>';
curl_close($ch);
```