Open Parcel Group API

The Open Parcel Group API provides direct access for customers to create and track Open Parcel Group shipments.

We've prepared step-by-step instructions so you can quickly integrate with Open Parcel Group, and start shipping right away.

Start with our step-by-step instructions to quickly get started.

API

Our REST based API allows you to be up and running faster than ever.

Working examples give your development team a head-start.

Step-By-Step

Our documentation, sample files, and Postman transactions reduce development time.

Performance

Our high-performance API will keep your operations team moving shipments.

Open Parcel Group API Basics

Shipment Workflow

For most API customers, the following shipment workflow will be appropriate
- Customer calls the Create Shipment API, specifying the shipper, consignee and shipment contents.
- We reply with:
  A shipping label.
  A tracking number that can be used with Tracking API, or on our Tracking Page
- Customer prints the shippling label and affixes it to their shipment.
- At end of day, Customer calls the Manifest API. All newly processed shipments are assigned to the manifest.
- We reply with a Manifest PDF
- Customer prints the Manfiest PDF, and sends it along with their shipments.
- A shipping label.
- A tracking number that can be used with Tracking API, or on our Tracking Page

End of Day Manifesting

A manifest groups parcels together, the grouping is used for tracking the parcels with a single manifest number

The manifest helps terminal operations perform a quick audit of the shipment contents, and scanning the manifest number allows a tracking event to be generated for every parcel in the shipment, even before they are individually scanned.

Creating Manifest Numbers

Manifest numbers are generated by customers, and are easy to create. They can be as simple as a date / time stamp in the format CCYYMMDD-HHMMSS. Customer may want to include other reference information in the manfiest number, like job number, cost center, or project identifier.

Customers can create manifests in three ways, pick the right method for your operations.

1) Pre-Assigned Manifest Numbers

For customers that generate a manifest number in advance, specify the job number in the "manifestnbr" field of the create shipment call. Finally call the manifest service with that same job number at the end of the day to complete the manifest.

2) End-of-Day Manifest Numbers

Optionally, create a manifest number at the end of the day, and include the number in the call to the manifest service.

When calling the manifest service in this manner, include the tracking number of every shipment in the manifest request.

3) Automatic Manifests

If you don't want to call the manifest service, all un-manifested shipments will be automatically assigned to a nightly manifest.

Tracking

Customers can track their shipments with either their shipment reference number, or the carrier tracking number returned from the Shipment API call.

Tracking data can be retreived in several ways:
- With the Tracking API. Use this to track individiual parcels or groups of undelivered parcels.
- With a tracking file, downloaded from the Open Parcel Group website.
- With a tracking file posted via SFTP to your web server.

If you're building your own database of tracking events, please don't use the individual tracking API calls. Use the FTP or batch tracking API, as documented in the trackign API service.

Message Formats - JSON and XML

The API supports both JSON and XML formats to support the widest variety of platforms.

Each web service is comprised of a **request** and **response** message.

All services use HTTPS POST (not GET) for communication.

For each API call, the caller prepares a request document (JSON or XML) and posts it to our API endpoint via HTTPS. All modern programming lanugages are supported (e.g. Java, C#, PHP, Ruby, JS/ES/TypeScript, etc.).

User Accounts

Logon Credentials

Use your existing Open Parcel Group credentials for all API calls.

Testing

Customers can begin development and testing right away with the following credentials:

URL	https://openparcelgroup.com/api
User ID	opg
Password	bettertogether

Open Parcel Group Online Security

SSL

All web services encrypt traffic with SSL to protect authentication credentials, shipper, and consignee data that is exchanged between customer applications and our web server.

Basic Authentication

Each web service is secured using standard HTTP Basic Authentication.

Basic Authentication is also easy to use with Postman, an outstanding tool for debugging web service calls.

Data Format and Data Types

JSON Data Types

The API conforms to the the following standards.
- Empty, optional, "null", or "false" attributes should not be included.

Each data field is configured with a data type, as outlined below.

Data Type	Description
Text	Most attribute values are text based. We validate incoming text attributes to not be null if required, and not more than then specified length.
Integer	A whole number starting at 0. If a decimal value is provide it will be rounded to the highest / lowest integer value withtout warning.
Decimal	A decimal number with up to 3 decimal places
Date Time	A date / time stamp combination, using the ISO 8601 standard in the format YYYY-MM-DDThh:mm:ss[.nnnnnnn][{+|-}hh:mm] For example, today's date 2025-08-31T06:03:09 NOTE: Time zone is optional, If the time zone is not included is assumed to be local time
Date	A date only, in the format "YYYY-MM-DD" For example, today's date 2025-08-31

Data Type	Description
Boolean	Use "1" or "true" to indicate that the flag / option is set to true, otherwise it's assumed to be false.
Base64	Used to embed a file like a PDF, Image, or ZPL file.

XML Data Types

The XML used in this API is attribute centric and conforms to the the following standards.
- All data is included in attributes, elements are used to group attributes
- Elements are self-closing, and only contain a closing tag if they contain a child element.
- Empty, optional, "null", or "false" attributes are not included. Don't set attributes to "" if they do not contain a value. False boolean attributes should not be included. Only include attributes if they are not null and if they contain a value.

Each data field is configured with a data type, as outlined below.

Data Type	Description
Text	Most attribute values are text based. We validate incoming text attributes to not be null if required, and not more than then specified length.
Integer	A whole number starting at 0. If a decimal value is provide it will be rounded to the highest / lowest integer value withtout warning.
Decimal	A decimal number with up to 3 decimal places
Date Time	A date / time stamp combination, using the ISO 8601 standard in the format YYYY-MM-DDThh:mm:ss[.nnnnnnn][{+|-}hh:mm] For example, today's date 2025-08-31T06:03:09 NOTE: Time zone is optional, If the time zone is not included is assumed to be local time
Date	A date only, in the format "YYYY-MM-DD" For example, today's date 2025-08-31
Boolean	1 indicates "true" / "yes". **NOTE:** Only include boolean attributes when they are true.
Base64	Used to embed a file like a PDF, Image, or ZPL file.

The shipment sevice is used to prepare each shipment / parcel. It's typically called first from software in the customer terminal, as part of a daily job. Send the shipment addressess (sender / consingnee) and parcel details for customs. The service returns a shipping label, cost estimate and tracking number.

Request Message
shipment

Represents a single shipment

Name	Description	Type	Optional	Sample Value
service	Specify the level of service for shipment	Text (50)		IN
test	Specify 1 to indicate that this is a test shipment	boolean	Optional	1

Name	Description	Type	Optional	Sample Value
reference	Customer reference code that uniquely idenfies this shipment. The value will be included in the shipping label and invoice data. Duplicate reference values will trigger an error.	Text (30)		123-12345
costcentre	Additional billing reference for charge-backs	Text (50)	Optional	Project 123
manifestnbr	A customer generated manifest number, used to group shipments together	Text (20)	Optional	20210617-065628
currency	Currency for shipment, item, and insured value. Use 3 character ISO 4217 Currency Code (ie USD, CAD, AUD, GBP, EUR)	Text (3)		CAD
value	Total value of all items, in the specified currency	Decimal (2)		10.99
weightunit	Unit of measure (LB or KG)	Text (2)		KG
weight	Total weight of the shipment, in the specified weight units	Decimal (4)		1.123
dimensionunit	Dimension Unit (IN = inches, CM = Centimeters) Defaults to CM if not specified	Text (2)	Optional	CM
labelformat	File format of the returned label. Valid values include ZPL or PDF. ZPL is used by default, if not specified.	Text (3)	Optional	PDF
tradeterms	Indicates duty payment terms. Valid values include DDU (Delivery Duty Unpaid) or DDP (Delivery Duty Paid)	Text (3)		DDU
exportype	Type of export, suggested values include "Permanent", "Temporary", "Return"	Text ()		

shipper

Shipper Address

Name	Description	Type	Optional	Sample Value
name	name of consignee, or company	Text (40)		ABC Company
attention	2nd, optional name / title / care of	Text (40)	Optional	Robert Dobolina
addr1	Address line 1	Text (40)		123 Main Street
addr2	Address line 2	Text (40)	Optional	Apartment 123
addr3	Address line 3	Text (40)	Optional	
city	City	Text (40)		New York
state	State / Province. Use standard postal abbreviations. Required for AU, US	Text (40)	Optional	NY
country	ISO 3166 Alpha 2 letter country code	Text (2)		US
postal	Postal Code	Text (12)		00123
phone	Local phone number	Text (20)	Optional	212 788 1400
email	email address, used for follow-up communication	Text (50)	Optional	support@deliverynet.ca

consignee

Consignee Address

Name	Description	Type	Optional	Sample Value
name	name of consignee, or company	Text (40)		ABC Company
attention	2nd, optional name / title / care of	Text (40)	Optional	Robert Dobolina
addr1	Address line 1	Text (40)		123 Main Street
addr2	Address line 2	Text (40)	Optional	Apartment 123
addr3	Address line 3	Text (40)	Optional	
city	City	Text (40)		New York
state	State / Province. Use standard postal abbreviations. Required for AU, US	Text (40)	Optional	NY
country	ISO 3166 Alpha 2 letter country code	Text (2)		US
postal	Postal Code	Text (12)		00123
phone	Local phone number	Text (20)	Optional	212 788 1400
email	email address, used for follow-up communication	Text (50)	Optional	support@deliverynet.ca

item

Include 1 item objest for each item in the shipment.

Name	Description	Type	Optional	Sample Value
item-code	Account Item identifier / SKU	Text (30)	Optional	SKU-123
description	Description of the item	Text (80)		iPad Case
qty	Quantity. The number of items, must be at least 1	int		5
unit	Type of item (box, pc, carton)	Text (25)	Optional	pc
value	Total value for all units (not the price of a single unit). If each unit costs 2.00, and quantiy=5, then indicate 10.00. Value must be in the same currency specified in the shipment	Decimal (2)		10.00
hscode	Harmonized system code	Text (6)		HS123
origin	Country of origin. Specify the ISO 3166-1 Alpha 2 Letter code for the country of origin	Text (2)	Optional	CN
export-reason	Identifies why the item is being exported (Sale, Gift, etc.)	Text (15)	Optional	Sale

package

Include 1 package object for each package in a shipment.

Name	Description	Type	Optional	Sample Value
reference	Package descriptor, or bar-code value affixed to the actual package.	Text (50)		
weight	Physical Weight, in the weight units specified	Decimal (3)		0.258

Name	Description	Type	Optional	Sample Value
length	Physical Length in the dimension units specified	Decimal (2)		25.0
width	Physical Width, in the dimension units specified	Decimal (2)		10.0
height	Physical Height, in the dimension units specified	Decimal (2)		2.5

Sample request, with return to shipper indicated. Don't specify a return address in this case.

```
{
    "shipment": {
    "service": "priority",
    "test": true,
    "account": "ABC",
    "datetime": "2019-06-18T13:04:53.295Z",
    "manifestnbr": "20210628-0228",
    "reference": "123-12345",
    "value": 10.00,
    "currency": "USD",
    "uom": "kg",
    "weight": 0.82,
    "originterminal": "YYZ",
    "shipper": {
        "name": "ABC Company",
        "attention": "Samuel Shipper",
        "addr1": "123 Main Street",
        "addr2": "Apartment 123",
        "city": "New York",
        "state": "NY",
        "country": "US",
        "postal": "3195",
        "phone": "212 234 1400",
        "email": "support@openparcelgroup.com"
        },
    "consignee": {
        "name": "Eiffel Tower",
        "attention": "Connie Consignee",
        "addr1": "Champ de Mars",
        "addr2": "5 Avenue Anatole",
        "city": "Paris",
        "state": "",
        "country": "FR",
        "postal": "75007",
        "phone": "212 234 1400",
        "email": "support@openparcelgroup.com"
        },
    "item": [
        {
```

```
            "description": "iPad",
            "qty": 5,
            "unit": "pc",
            "value": 10.00,
            "origin": "CN"
            }
        ]
    }
}
```

Sample request, with return to shipper indicated. Don't specify a return address in this case.

```xml
<xml>
 <shipment service="priority" test="1" account="ABC" datetime="2019-06-18T13:04:53.295Z"
reference="123-12345" value="10" currency="USD" uom="kg" weight="0.82" originterminal="YYZ">
    <shipper name="ABC Company" attention="Samuel Shipper" addr1="123 Main Street"
addr2="Apartment 123" city="New York" state="NY" country="US" postal="3195" phone="212 234 1400"
email="support@openparcelgroup.com" />
    <consignee name="Eiffel Tower" attention="Connie Consignee" addr1="Champ de Mars" addr2="5
Avenue Anatole" city="Paris" state="" country="FR" postal="75007" phone="212 234 1400"
email="support@openparcelgroup.com" />
    <item description="iPad" qty="5" unit="pc" value="10.00" origin="CN" />
 </shipment>
</xml>
```

Response Message
shipmentresponse

The shipment response is returned in response to the shipment request. Callers should first check for any error nodes before processing the response. The response contains a tracking number, cost estimate, and shipping label. Tracking is not available to all countries, tracking and trackingurl attributes will not be present if tracking is unavailable.

Name	Description	Type	Optional	Sample Value
tracknbr	Tracking Number that can be used for all subsequent tracking requests.Tracking not available to all countries, and will not be present when no tracking is available.	Text (30)	Optional	8675309
labelformat	PDF or ZPL returned. ZPL is not base 64 encoded. PDF is based 64 encoded.	Text (3)	Optional	ZPL
label	Shipping Label in the specified label format	base64		Base 64 File Content
costestimate	Shipping cost based on data provided and current rates.	Decimal (2)		10.23
costcurrency	Currency of cost estimate. ISO 4127 Currency Code, CAD, USD, EUR, AUD	Text (3)		CAD
trackingurl	Full URL for tracking, suitable for consignee use. The tracking URL embeds the tracknbr field. Tracking not available to all countries, and will not be present when no tracking is available.	Text (100)	Optional	

error

Name	Description	Type	Optional	Sample Value
code	Code value of the error	int		100
description	Text description of the error message	Text (500)		Invalid XML
context	Error context information	Text (50)	Optional	Error on line 1

Sample shipment response with ZPL label.

```
{
 "shipmentresponse": {
   "tracknbr": "1234567",
   "label": "^XA^CF,0,0,0^PR12^MD30^PW800^POI^CI13^LH0,20\n^FO12...Value Truncated"
 }
}
```

Sample shipment response with ZPL label.

```
<xml>
 <shipmentresponse tracknbr="1234567"
label="^XA^CF,0,0,0^PR12^MD30^PW800^POI^CI13^LH0,20&#xA;^FO12...Value Truncated" />
</xml>
```

Identifies a shipment to cancel, shipments must be cancelled before closing the manifest job.

Request Message

cancelshipment

Name	Description	Type	Optional	Sample Value
cancelnbr	Use either your customer reference number or the tracking number returned from the create shipment API call.	Text (50)		1234567890

Cancel shipment. Shipments may be cancelled as long as they have not been scanned.

Either tracking # or the reference # can be used to cancel shipment.

```
{
 "cancelshipment": {
   "cancelnbr": "1234567890"
 }
}
```

Cancel shipment. Shipments may be cancelled as long as they have not been scanned.

Either tracking # or the reference # can be used to cancel shipment.

```
<xml>
 <cancelshipment cancelnbr="1234567890" />
```

```
</xml>
```

Response Message

cancelshipmentresponse

Indicates the status of the cancelled shipment.

Name	Description	Type	Optional	Sample Value
cancelnbr	Tracking number assigned to the shipment	Text (50)		1234567890
status	"Success", if the cancellation was sucessful, otherwise see the error element.	Text (10)		Success

error

Any errors cancelling the shipment will appear in this element.

Name	Description	Type	Optional	Sample Value
code	Code value of the error	int		100
description	Text description of the error message	Text (500)		Invalid XML
context	Error context information	Text (50)	Optional	Error on line 1

Response

```
{
 "cancelshipmentresponse": {
   "tracknbr": "1234567890",
   "status": "Success"
 }
}
```

Response

```
<xml>
 <cancelshipmentresponse cancelnbr="1234567890" status="Success" />
</xml>
```

The manifest service is used to group inbound shipments. The manifest helps terminal operations verify reciept of the entire order, before scanning indivdiual pieces. Typically manifests are completed at the end of the day. The service returns a bar-coded PDF document that the terminal can use to scan the shipments.

Request Message

manifest

Used to close a manifest job, and retreive the manifest PDF document. The manifest will include all shipments since the last manifest call.

Name	Description	Type	Optional	Sample Value
manifestnbr	The unique manifest number to be closed.	Text (30)		12345

Name	Description	Type	Optional	Sample Value
	Use a date/time CCYYMMDDHHMM if you don't have one.			

Sample manifest, closes all shipments created with job number 12345

```
{
 "manifest": {
   "manifestnbr": "12345",
   "datetime": "2019-06-18T13:04:53.295Z"
 }
}
```

Sample manifest, creates job number 12345, and assigns shipments T1, T2, T3 to job number 12345.

```
{
 "manifest": {
   "manifestnbr": "12345",
   "datetime": "2017-07-18T13:04:53.295Z",
   "shipment": [
     {
       "tracknbr": "T1"
     },
     {
       "tracknbr": "T2"
     },
     {
       "tracknbr": "T3"
     }
   ]
 }
}
```

Sample manifest, closes all shipments created with job number 12345

```
<xml>
 <manifest manifestnbr="12345" datetime="2019-06-18T13:04:53.295Z" />
</xml>
```

Sample manifest, creates job number 12345, and assigns shipments T1, T2, T3 to job number 12345.

```
<xml>
 <manifest manifestnbr="12345" datetime="2017-07-18T13:04:53.295Z">
   <shipment tracknbr="T1" />
   <shipment tracknbr="T2" />
   <shipment tracknbr="T3" />
 </manifest>
</xml>
```

Response Message
manifestresponse

Contains the manifest PDF for the requested manifest job.

Name	Description	Type	Optional	Sample Value
manifestnbr	The manifest job number provided in the request.	Text (30)		12345
manifestpdf	Base 64 encoded PDF document describing the job, includes customer account information, total weight and shipment count.	base64		base 64 PDF content here...Base 64 File Content

Sample manifest response with Base 64 encoded PDF

```
{
 "manifestresponse": {
   "manifestnbr": "12345",
   "manifestpdf": "JVBERi0xLjUNJeLjz9MNCjcgMCBvYmoNPDwvTGluZWFyaXplZC...Value Truncated"
 }
}
```

Sample manifest response with Base 64 encoded PDF

```
<xml>
 <manifestresponse manifestnbr="12345"
manifestpdf="JVBERi0xLjUNJeLjz9MNCjcgMCBvYmoNPDwvTGluZWFyaXplZC...Value Truncated" />
</xml>
```

Used to track parcels.

Request Message
tracking

Specifies the tracking number of a specific number to track.

Name	Description	Type	Optional	Sample Value
tracknbr	Tracking number of the parcel to be tracked.	Text (50)		1234567890
type	Specify "new" to get all new tracking events received since the last tracking event.	Text (6)	Optional	new
account	3 letter account code, optional except for customers supporting multiple accounts	Text (3)	Optional	ABC

To maintain a local database, call the service with type='new' every 15 minutes.

```
{
 "tracking": {
   "type": "new",
```

```
    "account": "ABC"
  }
}
```

Use this for debugging and tracking a single shipment. NOTE: Single Shipment Tracking is limited to 60 calls / minute, it's a poor way to maintain a local database of tracking events.

```
{
 "tracking": {
    "tracknbr": "1234567890"
  }
}
```

To maintain a local database, call the service with type='new' every 15 minutes.

```
<xml>
 <tracking type="new" account="ABC" />
</xml>
```

Use this for debugging and tracking a single shipment. NOTE: Single Shipment Tracking is limited to 60 calls / minute, it's a poor way to maintain a local database of tracking events.

```
<xml>
 <tracking tracknbr="1234567890" />
</xml>
```

Response Message

trackingresponse

Tracking data for the specified tracking request. The tracking number can be either the reference number sent with your initial shipment request, or the carrier tracking number that is returned from the create shipment response.

Name	Description	Type	Optional	Sample Value
tracknbr	Tracking number for the shipment being tracked.	Text (50)		1234567890

trackingevent

Name	Description	Type	Optional	Sample Value
code	Tracking event code	Text (5)		FD01
description	Description of the event	Text (150)		Shipment Delivered
eventdate	Date and Time of the Tracking Event in the local time of the event.	datetime		2020-04-01T10:28
city	City, where the tracking event occurred.	Text (50)	Optional	London
country	Country code (ISO 3166) where the tracking event occurred.	Text (2)	Optional	GB

Tracking response with all "new" tracking events, for all shipments. The most recent event appears first.

```json
{
 "trackingresponse": [
   {
     "tracknbr": "1234567890",
     "trackingevent": [
       {
         "code": "510",
         "description": "Parcel Delivered",
         "eventdate": "2019-04-03T10:20",
         "city": "New York, NY",
         "country": "US"
       },
       {
         "code": "400",
         "description": "Cleared Customs",
         "eventdate": "2019-04-02T09:28",
         "city": "Buffalo, NY",
         "country": "US"
       },
       {
         "code": "300",
         "description": "Shipment Departed",
         "eventdate": "2019-04-01T13:28",
         "city": "Mississauga",
         "country": "CA"
       },
       {
         "code": "200",
         "description": "Shipment Scanned",
         "eventdate": "2019-04-01T12:28",
         "city": "Mississauga",
         "country": "CA"
       },
       {
         "code": "100",
         "description": "Shipment Data Received",
         "eventdate": "2019-04-01T09:28",
         "city": "Toronto",
         "country": "CA"
       }
     ]
   },
   {
     "tracknbr": "8675309",
     "trackingevent": [
       {
         "code": "200",
```

```
            "description": "Shipment Scanned",
            "eventdate": "2019-04-01T12:28",
            "city": "Mississauga",
            "country": "CA"
          },
          {
            "code": "100",
            "description": "Shipment Data Received",
            "eventdate": "2019-04-01T09:28",
            "city": "Toronto",
            "country": "CA"
          }
        ]
      }
    ]
  }
```

Tracking response for a single shipment, the most recent event appears first

```
{
  "trackingresponse": [
    {
      "tracknbr": "1234567890",
      "trackingevent": [
        {
          "code": "510",
          "description": "Parcel Delivered",
          "eventdate": "2019-04-03T10:20",
          "city": "New York, NY",
          "country": "US"
        },
        {
          "code": "400",
          "description": "Cleared Customs",
          "eventdate": "2019-04-02T09:28",
          "city": "Buffalo, NY",
          "country": "US"
        },
        {
          "code": "300",
          "description": "Shipment Departed",
          "eventdate": "2019-04-01T13:28",
          "city": "Mississauga",
          "country": "CA"
        },
        {
          "code": "200",
          "description": "Shipment Scanned",
          "eventdate": "2019-04-01T12:28",
          "city": "Mississauga",
```

```
              "country": "CA"
            },
            {
              "code": "100",
              "description": "Shipment Data Received",
              "eventdate": "2019-04-01T09:28",
              "city": "Toronto",
              "country": "CA"
            }
          ]
        },
        {
          "tracknbr": "8675309",
          "trackingevent": [
            {
              "code": "200",
              "description": "Shipment Scanned",
              "eventdate": "2019-04-01T12:28",
              "city": "Mississauga",
              "country": "CA"
            },
            {
              "code": "100",
              "description": "Shipment Data Received",
              "eventdate": "2019-04-01T09:28",
              "city": "Toronto",
              "country": "CA"
            }
          ]
        }
      ]
    }
```

Tracking response with all "new" tracking events, for all shipments. The most recent event appears first.

```
<xml>
 <trackingresponse tracknbr="1234567890">
    <trackingevent code="510" description="Parcel Delivered" eventdate="2019-04-03T10:20"
city="New York, NY" country="US" />
    <trackingevent code="400" description="Cleared Customs" eventdate="2019-04-02T09:28"
city="Buffalo, NY" country="US" />
    <trackingevent code="300" description="Shipment Departed" eventdate="2019-04-01T13:28"
city="Mississauga" country="CA" />
    <trackingevent code="200" description="Shipment Scanned" eventdate="2019-04-01T12:28"
city="Mississauga" country="CA" />
    <trackingevent code="100" description="Shipment Data Received" eventdate="2019-04-01T09:28"
city="Toronto" country="CA" />
 </trackingresponse>
 <trackingresponse tracknbr="8675309">
    <trackingevent code="200" description="Shipment Scanned" eventdate="2019-04-01T12:28"
```

```
city="Mississauga" country="CA" />
   <trackingevent code="100" description="Shipment Data Received" eventdate="2019-04-01T09:28"
city="Toronto" country="CA" />
 </trackingresponse>
</xml>
```

Tracking response for a single shipment, the most recent event appears first

```
<xml>
 <trackingresponse note="limited to 60 calls / minute" tracknbr="1234567890">
   <trackingevent code="510" description="Parcel Delivered" eventdate="2019-04-03T10:20"
city="New York, NY" country="US" />
   <trackingevent code="400" description="Cleared Customs" eventdate="2019-04-02T09:28"
city="Buffalo, NY" country="US" />
   <trackingevent code="300" description="Shipment Departed" eventdate="2019-04-01T13:28"
city="Mississauga" country="CA" />
   <trackingevent code="200" description="Shipment Scanned" eventdate="2019-04-01T12:28"
city="Mississauga" country="CA" />
   <trackingevent code="100" description="Shipment Data Received" eventdate="2019-04-01T09:28"
city="Toronto" country="CA" />
 </trackingresponse>
</xml>
```

Reference Values

The following values are used in the API calls.

Service

Use the following values in the Create Shipment API

Value	Description
Standard	Standard
Priority	Priority
Expedited	Expedited

Error Code List

This API will output a defined list of error codes.
- The current defined list of error codes and their associated descriptions appear on this list.

Number	Description
1001	The provided reference number is a duplicate.
1101	The weight provided is invalid
1102	Invalid Shipment
1103	An invalid customer account has been provided
1104	Shipment cannot be found
1105	Invalid ISO 3166-2 Country Code

Number	Description
1106	An error occurred creating the manifest
1107	No new tracking events are available.
1108	Unable to determine a rate for this shipment.
1109	No routing information found for this shipment
4000	Invalid JSON / XML
4001	Validation Error
8000	The API is currently unavailable
9000	An unexpected error has occurred.