

Oracle® Retail Collect and Receive Foundation Cloud Service Web Services Guide



G34954-01
July 2025



Oracle Retail Collect and Receive Foundation Cloud Service Web Services Guide,
G34954-01

Copyright © 2025, Oracle and/or its affiliates.

Primary Author: Gerlinde Rust

Contributing Authors:

Contributors:

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Send Us Your Comments

Preface

Audience	xviii
Documentation Accessibility	xviii
Customer Support	xviii

1 Implementation Overview

Collect and Receive Technical Overview	1-1
Collect and Receive Functional Integration	1-1

2 Implementation Considerations

Provisioning	2-1
--------------	-----

3 Get Started

Quick Start	3-1
Task 1: Obtain Account Information	3-1
Task 2: Obtain Access Token	3-1
Task 3: Send a Request	3-1
Authenticate	3-1
Send Requests	3-2
URL Structure	3-2
Supported Methods	3-2
Media Types	3-3
Supported Headers	3-3
Search Conventions	3-3
Where	3-3
OrderBy	3-4
Expands	3-4
Simple Field Comparison	3-4

Case Insensitive Comparison	3-5
Field Range Comparison	3-5
Compound Field Comparison	3-5
Nested Compound Field Comparison	3-5
Field to Field Comparison	3-6
Use Cases	3-6
Quote	3-6
Build	3-6
Get	3-8
List	3-9
Delivery Services	3-10
Build	3-10
Status	3-12
Cancel	3-13
Get	3-15
List	3-16

4 Collect and Receive Base APIs

Quote Services	4-2
Build a Quote — POST	4-2
Method	4-2
URL	4-2
Allowed Scope(s)	4-2
Request	4-3
Responses	4-9
Get a Quote — GET	4-17
Method	4-18
URL	4-18
Allowed Scope(s)	4-18
Request	4-18
Responses	4-18
Checks if a Quote Exists — HEAD	4-24
Method	4-24
URL	4-24
Allowed Scope(s)	4-24
Request	4-24
Responses	4-25
Queries for a Quote — GET	4-25
Method	4-25
URL	4-25
Allowed Scope(s)	4-25

Request	4-25
Responses	4-27
Queries for Quote Records/Count— GET	4-30
Method	4-30
URL	4-30
Allowed Scope(s)	4-30
Request	4-30
Responses	4-32
Queries for Quote Records by Attributes — GET	4-32
Method	4-32
URL	4-32
Allowed Scope(s)	4-32
Request	4-32
Responses	4-34
Delivery Services	4-36
Build a Delivery — POST	4-36
Method	4-37
URL	4-37
Allowed Scope(s)	4-37
Request	4-37
Responses	4-43
Cancel a Delivery — POST	4-58
Method	4-58
URL	4-58
Allowed Scope(s)	4-58
Request	4-58
Responses	4-59
Get the Current Status of a Delivery — GET	4-73
Method	4-73
URL	4-73
Allowed Scope(s)	4-73
Request	4-73
Responses	4-73
Get a Delivery — GET	4-79
Method	4-79
URL	4-80
Allowed Scope(s)	4-80
Request	4-80
Responses	4-80
Check if a Delivery Exists — HEAD	4-85
Method	4-85
URL	4-85

Allowed Scope(s)	4-85
Request	4-85
Responses	4-86
Queries for Delivery Records — GET	4-86
Method	4-86
URL	4-86
Allowed Scope(s)	4-86
Request	4-86
Responses	4-88
Queries for Delivery Records/Count — GET	4-93
Method	4-93
URL	4-93
Allowed Scope(s)	4-93
Request	4-93
Responses	4-95
Queries for Delivery Records by Attributes — GET	4-95
Method	4-95
URL	4-95
Allowed Scope(s)	4-95
Request	4-95
Responses	4-97
Metrics Services	4-102
Metrics — GET	4-102
Method	4-102
URL	4-103
Allowed Scope(s)	4-103
Request	4-103
Responses	4-103
Location Metrics — GET	4-105
Method	4-105
URL	4-105
Allowed Scope(s)	4-105
Request	4-106
Responses	4-106
Webhook Credential Services	4-108
Add Credential — POST	4-108
Method	4-108
URL	4-109
Allowed Scope(s)	4-109
Request	4-109
Responses	4-110
Get Credentials — GET	4-112

Method	4-112
URL	4-112
Allowed Scope(s)	4-112
Request	4-112
Responses	4-112
Get Credential — GET	4-113
Method	4-113
URL	4-114
Allowed Scope(s)	4-114
Request	4-114
Responses	4-114
Update Credential — PUT	4-116
Method	4-116
URL	4-117
Allowed Scope(s)	4-117
Request	4-117
Responses	4-118
Response Code: 400	4-119
Response Code: 404	4-120
Response Code: 500	4-120
Delete Credential — DELETE	4-120
Method	4-120
URL	4-120
Allowed Scope(s)	4-120
Request	4-121
Responses	4-121
Status Notification Webhook Services	4-122
Leveraging the Status Notification Webhook for Delivery Communications	4-125
Add a New Webhook — POST	4-126
Method	4-126
URL	4-127
Allowed Scope(s)	4-127
Request	4-127
Responses	4-128
Queries for Webhook Records — GET	4-130
Method	4-130
URL	4-131
Allowed Scope(s)	4-131
Request	4-131
Responses	4-133
Performs a Bulk Action using a Collection of Webhook — POST	4-134
Method	4-134

URL	4-135
Allowed Scope(s)	4-135
Request	4-135
Responses	4-136
Queries for Webhook Records/Count — GET	4-138
Method	4-138
URL	4-138
Allowed Scope(s)	4-139
Request	4-139
Responses	4-141
Get a Webhook — GET	4-141
Method	4-141
URL	4-141
Allowed Scope(s)	4-141
Request	4-141
Responses	4-142
Update a Webhook — PUT	4-143
Method	4-143
URL	4-143
Allowed Scope(s)	4-143
Request	4-143
Responses	4-145
Delete a Webhook — DELETE	4-147
Method	4-147
URL	4-148
Allowed Scope(s)	4-148
Request	4-148
Responses	4-148
Checks If a Webhook Exists — HEAD	4-148
Method	4-148
URL	4-149
Allowed Scope(s)	4-149
Request	4-149
Responses	4-149
Queries for Webhook Records By Attributes — GET	4-149
Method	4-149
URL	4-150
Allowed Scope(s)	4-150
Request	4-150
Responses	4-151
Data Retention Configuration Service	4-152
Add a New DataRetentionConfig — POST	4-152

Method	4-152
URL	4-152
Allowed Scope(s)	4-152
Request	4-152
Responses	4-154
Queries for DataRetention Configuration Records — GET	4-156
Method	4-156
URL	4-157
Allowed Scope(s)	4-157
Request	4-157
Responses	4-159
Queries for DataRetention Configuration Records/Count — GET	4-160
Method	4-160
URL	4-161
Allowed Scope(s)	4-161
Request	4-161
Responses	4-163
Get a DataRetention Configuration — GET	4-163
Method	4-163
URL	4-163
Allowed Scope(s)	4-163
Request	4-163
Responses	4-164
Update a DataRetention Configuration — PUT	4-165
Method	4-165
URL	4-165
Allowed Scope(s)	4-165
Request	4-165
Responses	4-167
Delete a DataRetention Configuration — DELETE	4-169
Method	4-169
URL	4-170
Allowed Scope(s)	4-170
Request	4-170
Responses	4-170
Checks If a DataRetention Configuration Exists — HEAD	4-170
Method	4-170
URL	4-171
Allowed Scope(s)	4-171
Request	4-171
Responses	4-171
Queries for DataRetention Configuration Records By Attributes — GET	4-171

Method	4-171
URL	4-172
Allowed Scope(s)	4-172
Request	4-172
Responses	4-172
Error Handling	4-174
Bean Validations	4-174
Error Representation	4-174
Error Codes	4-174
Collect and Receive Error Codes	4-175

5 Foundation Service

Location Service	5-1
Add a New Location — POST	5-1
Method	5-1
URL	5-1
Allowed Scope(s)	5-2
Request	5-2
Responses	5-4
Queries for Location Records — GET	5-7
Method	5-7
URL	5-7
Allowed Scope(s)	5-7
Request	5-7
Responses	5-9
Performs a Bulk Action Using a Collection of Locations — POST	5-11
Method	5-11
URL	5-11
Allowed Scope(s)	5-11
Request	5-11
Responses	5-13
Queries for Location Records/Count — GET	5-17
Method	5-17
URL	5-17
Allowed Scope(s)	5-17
Request	5-17
Responses	5-18
Get a Location — GET	5-19
Method	5-19
URL	5-19
Allowed Scope(s)	5-19

Request	5-19
Responses	5-20
Update a Location — PUT	5-22
Method	5-22
URL	5-22
Allowed Scope(s)	5-22
Request	5-22
Responses	5-24
Delete a Location — DELETE	5-27
Method	5-27
URL	5-28
Allowed Scope(s)	5-28
Request	5-28
Responses	5-28
Checks if a Location Exists — HEAD	5-28
Method	5-28
URL	5-29
Allowed Scope(s)	5-29
Request	5-29
Responses	5-29
Queries for Location Records by Attributes— GET	5-29
Method	5-29
URL	5-30
Allowed Scope(s)	5-30
Request	5-30
Responses	5-31
Package Dimension Service	5-33
Add a New PackageDimension — POST	5-33
Method	5-33
URL	5-34
Allowed Scope(s)	5-34
Request	5-34
Responses	5-34
Queries for PackageDimension Records— GET	5-37
Method	5-37
URL	5-38
Allowed Scope(s)	5-38
Request	5-38
Responses	5-40
Performs a Bulk Action Using a Collection of PackageDimension — GET	5-42
Method	5-42
URL	5-42

Allowed Scope(s)	5-42
Request	5-42
Responses	5-44
Queries for PackageDimension Records/Count— GET	5-48
Method	5-48
URL	5-48
Allowed Scope(s)	5-48
Request	5-48
Responses	5-49
Get a PackageDimension — GET	5-50
Method	5-50
URL	5-50
Allowed Scope(s)	5-50
Request	5-50
Responses	5-50
Update a PackageDimension — PUT	5-52
Method	5-52
URL	5-52
Allowed Scope(s)	5-53
Request	5-53
Responses	5-54
Delete a PackageDimension — DELETE	5-58
Method	5-58
URL	5-58
Allowed Scope(s)	5-58
Request	5-58
Responses	5-58
Checks if a PackageDimension Exists — HEAD	5-59
Method	5-59
URL	5-59
Allowed Scope(s)	5-59
Request	5-59
Responses	5-59
Queries for PackageDimension Records by Attributes — GET	5-59
Method	5-60
URL	5-60
Allowed Scope(s)	5-60
Request	5-60
Responses	5-61
Reason Code Service	5-63
Add a New Reason — POST	5-63
Method	5-63

URL	5-63
Allowed Scope(s)	5-64
Request	5-64
Responses	5-65
Queries for Reason Records— GET	5-68
Method	5-68
URL	5-68
Allowed Scope(s)	5-68
Request	5-68
Responses	5-70
Performs a Bulk Action Using a Collection of Reason — GET	5-71
Method	5-71
URL	5-72
Allowed Scope(s)	5-72
Request	5-72
Responses	5-73
Queries for Reason Records/Count— GET	5-77
Method	5-77
URL	5-77
Allowed Scope(s)	5-77
Request	5-77
Responses	5-78
Get a Reason — GET	5-79
Method	5-79
URL	5-79
Allowed Scope(s)	5-79
Request	5-79
Responses	5-79
Update a Reason — PUT	5-81
Method	5-81
URL	5-81
Allowed Scope(s)	5-81
Request	5-81
Responses	5-83
Delete a Reason — DELETE	5-85
Method	5-85
URL	5-86
Allowed Scope(s)	5-86
Request	5-86
Responses	5-86
Checks if a Reason Exists — HEAD	5-86
Method	5-86

URL	5-87
Allowed Scope(s)	5-87
Request	5-87
Responses	5-87
Queries for Reason Records by Attributes — GET	5-87
Method	5-87
URL	5-88
Allowed Scope(s)	5-88
Request	5-88
Responses	5-88

6 User Service

currentUser — GET	6-1
Method	6-1
URL	6-1
Request	6-1
Responses	6-1
Response Code: 200	6-1
Reason Code: 400	6-2
Reason Code: 403	6-3
Reason Code: 500	6-3
updateUser — PUT	6-3
Method	6-4
URL	6-4
Request	6-4
Example	6-4
User — Schema	6-4
Responses	6-5
Response Code: 204	6-5
Reason Code: 400	6-5
Reason Code: 500	6-6

7 Administration Service

createClient — POST	7-1
Method	7-1
URL	7-1
Allowed Scope(s)	7-1
Request	7-1
Example Value	7-1
Schema — Client Request	7-1

Responses	7-2
Response Code: 200	7-2
Response Code: 400	7-2
Response Code: 424	7-4
Response Code: 500	7-4
forgetPrivateData — POST	7-4
Method	7-4
URL	7-4
Allowed Scope(s)	7-4
Request	7-4
Responses	7-5
Response Code: 200	7-5
Response Code: 204	7-5
Response Code: 400	7-5
Response Code: 409	7-6
Response Code: 500	7-6
getPrivateData — GET	7-6
Method	7-6
URL	7-6
Allowed Scope(s)	7-6
Request	7-6
Responses	7-7
Response Code: 200	7-7
Response Code: 400	7-8
Response Code: 500	7-9
addScopes — PUT	7-9
Method	7-9
URL	7-9
Allowed Scope(s)	7-9
Request	7-9
Responses	7-10
Response Code: 204	7-10
Response Code: 400	7-10
Response Code: 404	7-10
Response Code: 424	7-10
Response Code: 500	7-10
getClients— GET	7-11
Method	7-11
URL	7-11
Allowed Scope(s)	7-11
Request	7-11
Responses	7-11

Response Code: 200	7-11
Response Code: 404	7-12
Response Code: 424	7-12
Response Code: 500	7-12

8 Version Updates

Send Us Your Comments

Oracle® Retail Collect and Receive Foundation Cloud Service Web Services Guide

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document.

Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).



Note:

Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the Online Documentation available on the Oracle Technology Network Web site. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: retail-doc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at <http://www.oracle.com>.

Preface

The *Oracle® Retail Collect and Receive Foundation Cloud Service Web Services Guide* describes all available functions.

Audience

This Web Services Guide is for System Administrators and Operations personnel, as well as Integrators and implementation staff personnel.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Customer Support

To contact Oracle Customer Support, access My Oracle Support at the following URL:

<https://support.oracle.com>

When contacting Customer Support, please provide the following:

- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create
- Exact error message received
- Screen shots of each step you take

1

Implementation Overview

The purpose of this document is to provide administration tasks and integration information for Oracle Retail Collect and Receive Foundation Cloud Service.

This cloud service provides premier delivery options, bringing together Oracle's industry IP and cloud infrastructure with Uber's extensive network of drivers and cutting-edge logistics technology to offer reliable and cost-effective same-day delivery services. Whether bookings are taken on the website, contact center, or in the retail store, Collect and Receive facilitates the booking of the Uber courier for the delivery of the package to the consumer or for pickup up of a return from the consumer to deliver to the drop off location. Same Day Delivery and Same Day Returns transform the Omnichannel experience by providing businesses with a fast, efficient, and cost-effective way to meet customers' growing demand for faster delivery options.

Collect and Receive Technical Overview

Collect and Receive is a cloud native solution leveraging the most recent and advanced technologies used in Oracle Cloud applications architecture including Autonomous Database.

Collect and Receive is a Java-based application deployed on Oracle Cloud Infrastructure. The application is deployed in a highly available, high performance, scalable architecture.

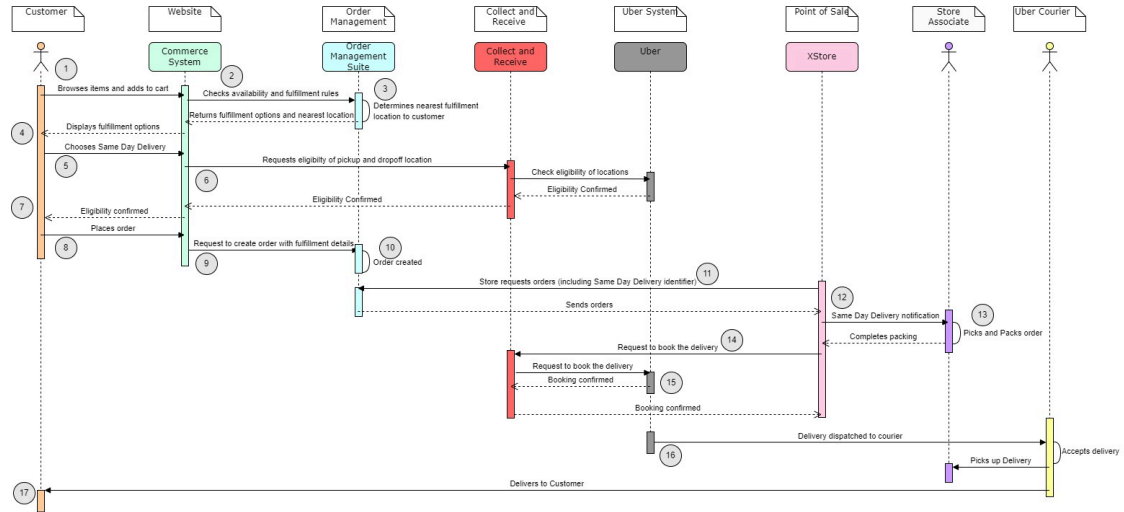
Key technical features include the following:

- Browser-based User Interface for scheduling and managing deliveries
- Application APIs for external systems
- Same Day Delivery services using Uber's Delivery as a Service (DaaS) API
- OAuth 2.0 based authentication via Oracle Cloud Infrastructure Identity and Access Management (OCI IAM)
- Internal Job Scheduler
- Event-driven webhook notifications to receive near real-time deliver status notifications

Collect and Receive Functional Integration

The diagrams below provide a sample functional overview for how Collect and Receive integrates with other systems. These diagrams highlight just one example of how these journeys can be architected, though there are other possible journeys and touch point considerations to craft the desired customer experience. The first diagram illustrates a Same Day Delivery journey (Store to Customer) and the second illustrates a Returns journey (Customer to Store). The accompanying explanations are written from a system-to-system perspective, noting steps in the process and the movement of data associated with those steps.

Figure 1-1 Functional Integration — Store to Customer

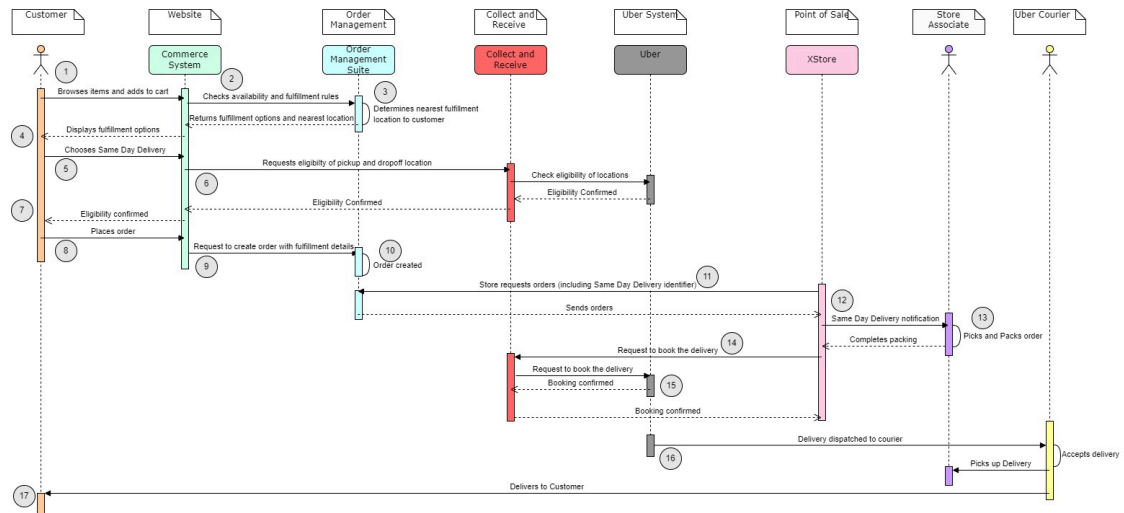


Store to Customer (Same Day Delivery)

Use Case: Customer purchases via Website for Same Day Delivery, fulfilled from nearby Store

1. Customer browses items on the Website and adds them to the cart.
2. Website checks item availability and fulfillment rules with Order Management System.
3. Order Management System determines the nearest fulfillment location to the Customer.
4. Website displays the fulfillment options, including Same Day Delivery.
5. Customer selects Same Day Delivery.
6. Website calls Collect and Receive, which calls Uber, to check eligibility of delivery between Store and Customer locations.
7. Eligibility is confirmed to the Customer.
8. Customer places the order.
9. Website books order with the Order Management System, which includes fulfillment type.
10. Order Management System creates the order.
11. Point of Sale System requests orders to fulfill from Order Management System.
12. Point of Sale System notifies Store Associates of Same Day Delivery orders.
13. Store Associate picks and packs the order.
14. Point of Sale System calls Collect and Receive, which calls Uber, to book the delivery.
15. Delivery is created successfully.
16. Uber dispatches Courier to Store to pick up the order and then deliver to the Customer.
17. Customer receives delivery.

Figure 1-2 Functional Integration — Customer to Store (Returns)



Customer to Store (Returns)

Use Case: Customer initiates request for Return Authorization (RA) via Website

1. Customer initiates request for a Return Authorization for their order on the Website.
2. Website calls the Order Management System to get back the eligible return items on the order.
3. Item(s) eligible for return are displayed on the Website to the Customer.
4. Customer selects item(s) to return and elects for a Courier to pick up their return item(s).
5. Website requests a Return Authorization to be created by the Order Management System.
6. Order Management System creates the Return Authorization and determines the Store nearest to the Customer's address.
7. Order Management System calls Collect and Receive, which calls Uber, to check eligibility of a delivery between the Customer and Store locations.
8. Eligibility is confirmed and the Website offers the option to schedule a courier to pick up the return items.
9. Customer schedules a courier pickup.
10. The Website calls the Order Management System, which calls Collect and Receive, which calls Uber to book the return pickup.
11. Returns pickup is confirmed. Confirmation displays on Website to the Customer.
12. Uber dispatches Courier to the Customer to pick up the Returns and then deliver to the Store.
13. On receipt, the Store Associate scans the package label (Delivery ID) at the Point of Sale to look up the Return Authorization and order.
14. The Point of Sale System calls the Order Management System to request the Return Authorization details.
15. The Point of Sale System displays the Return Authorization details to the Store Associate.
16. Store Associate confirms receipt of return item(s) into Point of Sale.

17. The Point of Sale System sends a message to the Order Management System that the Return Authorization items have been received.
18. Customer receives confirmation of processed return.

2

Implementation Considerations

This chapter contains implementation considerations when implementing Collect and Receive.

Provisioning

This section describes high-level provisioning steps for Collect and Receive.

In an implementation that includes Collect and Receive, the following steps apply before the system is running.

- **Uber Setup**
The customer must work directly with Uber to set up a Delivery as a Service (DaaS) account.
- **Provisioning**
Provisioning includes the setup and configuration for this account within the Collect and Receive. Uber will contact Oracle as needed to complete the setup.
- **Access Setup**
The customer will assign users and create clients (for direct web service access) as needed.

3

Get Started

Quick Start

Complete these tasks to set up your environment and use Collect and Receive.

Task 1: Obtain Account Information

From your account administrator, obtain the applicable account credentials to enable you to access Collect and Receive functions and resources.

Specifically, you will need one of the following:

- User name and password
- Client ID and secret

Task 2: Obtain Access Token

After you set up your REST client, follow steps in Authenticate section to obtain an access token in response.

Task 3: Send a Request

You can send a request to ensure that your connection works, for example, retrieving a count of orders.

```
curl -i -X GET -H "Authorization:Bearer <access-token>" https://  
<HOSTNAME>:<PORT>/<service>/api/delivery/v2/delivery/count
```

If the request is successful, you should receive a 200 response with a body such as the following:

```
50
```

Congratulations!

Authenticate

Collect and Receive uses OAuth 2.0 protocol for authentication, enabling clients to connect securely to the server using Client Credential authorization grant.

**Note:**

Ensure that you have the appropriate authentication credentials.

You access the Collect and Receive REST resources over HTTPS and you must provide the following information for authentication:

- User name and password for your Collect and Receive or an appropriate application client_id and secret.
- The client's scope.
- The authorization grant type, which is client_credentials.

For example, to authenticate using cURL:

- Construct a POST request using the -X POST cURL option.
- Prepare the request body encoded in application/x-www-form-urlencoded content type using -d cURL option.
- Pass the required credential, scope, and grant type.

The following provides an example cURL command:

```
curl -i -X POST -u <client id>:<client secret> -H "Content-Type:
application/x-www-form-urlencoded" -d
"grant_type=client_credentials&scope=<scope>" https://<host>:<PORT>/oauth2/v2/
token
```

Send Requests

Use these guidelines when sending requests using the Collect and Receive REST API.

URL Structure

Here's the URL structure for the requests:

```
https://<HOSTNAME>:<PORT>/<service>/<resource-path>
```

Where:

- <HOSTNAME>:<PORT>: Host and port where Collect and Receive is running.
- <service>: The service identifier.<resource-path>: Relative path that defines the resource.

Supported Methods

You can perform basic CRUD operations (create, read, update, and delete) on a resource by using standard HTTP method requests, as summarized in the following table.

Table 3-1 HTTP Method Requests

HTTP Method	Description
GET	Retrieve information about the resource.
POST	Create the resource.
PUT	Update the resource.
DELETE	Delete the resource.

Media Types

The following media types are supported by the Collect and Receive REST API:

- application/json

Supported Headers

The Collect and Receive REST API supports the following headers that may be passed in the header section of the HTTP request or response.

Table 3-2 Headers

Header	Description	Example
Authorization	Supply the bearer token to access OAuth 2.0 protected resources.	Authorization: Bearer <access-token>
Accept	Indicate the content type the client is able to understand.	Accept: application/json
Content-Type	Media type of the body of the request/response.	Content-Type: application/json

Search Conventions

This section describes the Search Conventions.

Where

The where parameter is used to filter results within list and count requests. It is a JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource. For example, a delimited field named child.id would filter by the entity field of this resource named child and that entity has a field named id.

Querying is supported in three ways:

- If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or "startsWith" comparison. For

example, ``{"child.name":"Jo"}`` would match Joe, John, and so on. For all other types, it serves as the lower bound of a `>=` comparison, `{"minAmount":100.00}`.

- If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example `{"id":[1, 2, 3]}`.
- If an object is provided, a variety of options are supported:
 - Fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example `{"id":{"from":0,"to":100}}`
 - A field named `singleValue` can be used to follow the single value pattern discussed above.
 - A field named `in` can be used to follow the array pattern discussed above.
 - A field named `fieldPath` can be used to specify the path of a related field.
 - A field named `caseInsensitive` can be used to enable case-insensitive querying of character-based fields.
 - A field named `invert` can be used to invert a predicate (that is as a NOT).

Combining criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with `@and` and `@or` with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple `@and` / `@or` groupings within a given object, take care to suffix them with additional characters (that is `@and1` , `@and2` , and so on) to ensure that they are uniquely named since any duplicates will be eliminated.

Some examples, which may be helpful when using this functionality, can be found at Query Examples.

OrderBy

A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria.

- When supplied as strings, names are provided in the same format as the where argument - optionally followed by ``asc`` (default) or ``desc`` to control order direction.
- When supplied as a JSON array, a String `fieldName` field must be included in the object. Optional String `order` (asc or desc) and boolean `caseInsensitive` fields may also be provided.

Expands

A comma-delimited list of relationship paths, in dot-notated path form (that is, `child.line` would return the relationship named `child`, and its `child` relationship named `line`), on the resulting model which should be included in the response or `all` if all associated relationships should be included. By default, all `*:1` relationships are always present in service responses. Any `*:Many` relationships are always excluded by default, and the expands mechanism is the means by which callers may request for these relationships to be returned in service responses.

Simple Field Comparison

In this example we look for records which have an id ≥ 1 .

```
{
  "id": 1
}
```

Case Insensitive Comparison

In this example we look for records which have a reference id like AAA%, but ignore case.

```
{
  "deliveryId": {
    "singleValue": "aaa",
    "caseInsensitive": true
  }
}
```

Field Range Comparison

In this example, we look for records with a referenceId between AAA and ZZZZZZZZZZZZZZ.

```
{
  "deliveryId": {
    "from": "AAA",
    "to": "ZZZZZZZZZZZZZZ"
  }
}
```

Compound Field Comparison

In this example, we look for records with a crossReference with a key of TEST_TRANSACTION_ID and value of 12345.

```
{
  "crossReferences.key": {
    "in": ["TEST_TRANSACTION_ID"]
  },
  "crossReferences.value": {
    "in": ["12345"]
  }
}
```

Nested Compound Field Comparison

In this example, we look for fields with a reference Id \geq AAA1111 and a crossReference with a key of TEST_TRANSACTION_ID and value of 12345.

```
{
  "deliveryId": "AAA1111",
```

```
"@and1": {
  "crossReferences.key": {
    "in": ["TEST_TRANSACTION_ID"]
  },
  "crossReferences.value": {
    "in": ["12345"]
  }
}
```

Field to Field Comparison

In this example we look for records where deliveryId is not equal to the currency, ignoring the case of the two.

```
{
  "deliveryId": {
    "fieldPath": "currency",
    "caseInsensitive": true,
    "invert": true
  }
}
```

Use Cases

Use cases demonstrate how to make use of REST APIs, including how to create and update records, and more. Use them to understand quickly how REST APIs can meet your business needs.

To get the most benefit from use cases, you should be familiar with basic REST concepts discussed in the [Quick Start](#).

Quote

This section explains how to make use of Quote-related functions. Quotes are a mechanism by which eligibility for same-day delivery and estimated costs can be determined.

Build

To submit a request to obtain one or more quotes, the build resource method is provided. This method accepts a delivery request, and returns 0 or more quotes from eligible providers.

POST

`https://<HOSTNAME>:<PORT>/<service>/api/delivery/v2/quote/build`

Example Request

The following is an example of the request body in JSON format.

```
{
  "deliveryId": "0000001",
  "totalValue": 100.00,
  "pickupContact": {
    "companyName": "<COMPANYNAME>",
    "firstName": "<FIRSTNAME>",
    "lastName": "<LASTNAME>",
    "phoneNumber": "<PHONENUMBER>"
  },
  "pickupAddress": {
    "address1": "<ADDRESS>",
    "city": "<CITY>",
    "state": "<STATE>",
    "postalCode": "<POSTALCODE>",
    "country": "<COUNTRY>"
  },
  "dropoffContact": {
    "firstName": "<FIRSTNAME>",
    "lastName": "<LASTNAME>",
    "phoneNumber": "<PHONENUMBER>"
  },
  "dropoffAddress": {
    "address1": "<ADDRESS>",
    "city": "<CITY>",
    "state": "<STATE>",
    "postalCode": "<POSTALCODE>",
    "country": "<COUNTRY>"
  },
  "manifest": [
    {
      "description": "Package 1",
      "identifier": "<ID>",
      "items": [
        {
          "description": "widget",
          "identifier": "<ID>",
          "quantity": 2
        }
      ],
      "length": 5,
      "height": 5,
      "width": 5,
      "weight": 10
    }
  ]
}
```

Example Response

The following is an example of a successful response body in JSON format.

```
{
  "quotes": [
    {
```

```

        "created": "2023-06-02T13:54:39.346Z",
        "createdBy": "<UID>",
        "id": 1,
        "crossReferences": [],
        "currency": "USD",
        "deliveryId": "0000001",
        "dropoffDeadline": "2023-06-02T15:24:14.000Z",
        "dropoffEta": "2023-06-02T14:53:48.000Z",
        "expires": "2023-06-02T14:09:39.241Z",
        "fee": 13.6,
        "provider": "<PROVIDER>",
        "providerCreated": "2023-06-02T13:54:39.241Z",
        "providerInstance": "<PROVIDERINSTANCE>",
        "providerId": "<PROVIDERID>"
    }
}
]
}

```

Other Responses

In the event of the request not succeeding, a number of other responses may be returned.

Table 3-3 Other Responses — Request Not Successful

HTTP Code	Explanation	Body
204	No providers were eligible to service this request.	NA
400	The provided request input was not valid.	An array of MapperErrorData objects describing issues with the request.
404	A provided identifier or provider could not be found.	NA
424	One or more eligible providers were identified, but none were able to service the request.	An array of ProviderError objects explaining the issue(s).
500	The request failed for an unknown reason.	NA

Get

To retrieve a previously generated quote, the get method is provided. This method retrieves a single quote using its (id) identifier.

GET

`https://<HOSTNAME>:<PORT>/<service>/api/delivery/v2/quote/{id}`

Example Response

The following is an example of a successful response body in JSON format.

```

"created": "2023-06-02T13:54:39.346Z",

```

```
"createdBy": "<UID>",
"id": <ID>,
"crossReferences": [],
"currency": "USD",
"deliveryId": "0000001",
"dropoffDeadline": "2023-06-02T15:24:14.000Z",
"dropoffEta": "2023-06-02T14:53:48.000Z",
"expires": "2023-06-02T14:09:39.241Z",
"fee": 13.6,
"provider": "<PROVIDER>",
"providerCreated": "2023-06-02T13:54:39.241Z",
"providerInstance": "<PROVIDERINSTANCE>",
"providerId": "<PROVIDERID>"
}
```

Other Responses

In the event of the request not succeeding, a number of other responses may be returned.

Table 3-4 Other Responses — Request Not Successful

HTTP Code	Explanation
404	The requested quote could not be found.
500	The request failed for an unknown reason.

List

To obtain a list of quotes, the list method is provided. This method returns an array of quotes matching the provided criteria. See [Quick Start](#) for more information on this endpoint's available options.

GET

```
https://<HOSTNAME>:<PORT>/<service>/api/delivery/v2/quote?where={"deliveryId":
["0000001"]}
```

Example Response

The following is an example of a successful response body in JSON format.

```
[
  {
    "created": "2023-06-02T13:54:39.346Z",
    "createdBy": "<UID>",
    "id": 1,
    "crossReferences": [],
    "currency": "USD",
    "deliveryId": "0000001",
    "dropoffDeadline": "2023-06-02T15:24:14.000Z",
    "dropoffEta": "2023-06-02T14:53:48.000Z",
    "expires": "2023-06-02T14:09:39.241Z",
    "fee": 13.6,
```



```
        "provider": "<PROVIDER>",
        "providerCreated": "2023-06-02T13:54:39.241Z",
        "providerInstance": "<PROVIDERINSTANCE>",
        "providerId": "<PROVIDERID>"
    }
}
```

Other Responses

In the event of the request not succeeding, a number of other responses may be returned.

Table 3-5 Responses — Request Not Successful

HTTP Code	Explanation
500	The request failed for an unknown reason.

Delivery Services

This section explains how to make use of Delivery-related functions. Deliveries represent successfully arranged same-day delivery orders.

Build

To submit a request to obtain a delivery, the build resource method is provided. This method accepts a delivery request, and returns a single delivery from an eligible provider.

POST

`https://<HOSTNAME>:<PORT>/<service>/api/delivery/v2/delivery/build`

Example Request

The following is an example of the request body in JSON format.

```
{
  "deliveryId": "0000001",
  "totalValue": 100.00,
  "pickupContact": {
    "companyName": "<COMPANYNAME>",
    "firstName": "<FIRSTNAME>",
    "lastName": "<LASTNAME>",
    "phoneNumber": "<PHONENUMBER>"
  },
  "pickupAddress": {
    "address1": "<ADDRESS>",
    "city": "<CITY>",
    "state": "<STATE>",
    "postalCode": "<POSTALCODE>",
    "country": "<COUNTRY>"
  },
}
```

```

"dropoffContact": {
  "firstName": "<FIRSTNAME>",
  "lastName": "<LASTNAME>",
  "phoneNumber": "<PHONENUMBER>"
},
"dropoffAddress": {
  "address1": "<ADDRESS>",
  "city": "<CITY>",
  "state": "<STATE>",
  "postalCode": "<POSTALCODE>",
  "country": "<COUNTRY>"
},
"manifest": [
  {
    "description": "Package 1",
    "identifier": "0000001 -01-01",
    "items": [
      {
        "description": "widget",
        "identifier": "<ID>",
        "quantity": 2
      }
    ],
    "length": 5,
    "height": 5,
    "width": 5,
    "weight": 10
  }
]

```

Example Response

The following is an example of a successful response body in JSON format.

```

{
  "delivery": {
    "created": "2023-06-02T15:58:14.260Z",
    "createdBy": "UID",
    "id": 1,
    "barcodes": [
      {
        "created": "2023-06-02T15:58:14.298Z",
        "createdBy": "<UID>",
        "id": 1,
        "identifier": "<ID>",
        "parentId": <PARENTID>,
        "type": "QR",
        "value": "0000001 -01-01"
      }
    ],
    "crossReferences": [],
    "currency": "USD",
    "deliveryId": "<DELIVERYID>",
    "dropoffDeadline": "2023-06-02T17:33:50.000Z",
    "dropoffEta": "2023-06-02T17:00:07.000Z",
  }
}

```

```

        "fee": 13.6,
        "manifestSize": 1,
        "pickupEta": "2023-06-02T16:18:14.000Z",
        "provider": "<PROVIDER>",
        "providerCreated": "2023-06-02T15:58:13.965Z",
        "providerInstance": "<PROVIDERINSTANCE>",
        "providerId": "<PROVIDERVID>",
        "status": "PENDING",
        "statusHistory": [],
        "trackingURL": "https://<PROVIDER>.com/path/to/tracking/url"
    }
}

```

Other Responses

In the event of the request not succeeding, a number of other responses may be returned.

Table 3-6 Other Responses — Request Not Successful

HTTP Code	Explanation	Body
204	No providers were eligible to service this request.	NA
400	The provided request input was not valid.	An array of MapperErrorData objects describing issues with the request.
404	A provided identifier or provider could not be found.	NA
409	The provided deliveryId already exists.	NA
424	One or more eligible providers were identified, but none were able to service the request.	An array of ProviderError objects explaining the issue(s).
500	The request failed for an unknown reason.	NA

Status

To retrieve a previously arranged delivery, reflecting its current to-the-minute status, the status endpoint is provided. This method retrieves a single delivery using either its {id} or {deliveryId}.

GET

`https://<HOSTNAME>:<PORT>/<service>/api/delivery/v2/delivery/status?id={id}`

`https://<HOSTNAME>:<PORT>/<service>/api/delivery/v2/delivery/status?
deliveryId={deliveryId}`

Example Response

The following is an example of a successful response body in JSON format.

```
{
  "created": "2023-06-02T15:58:14.260Z",
  "createdBy": "<UID>",
  "id": 1,
  "barcodes": [
    {
      "created": "2023-06-02T15:58:14.298Z",
      "createdBy": "<UID>",
      "id": 1,
      "identifier": "<ID>",
      "parentId": <PARENTID>,
      "type": "QR",
      "value": "0000001 -01-01"
    }
  ],
  "crossReferences": [],
  "currency": "USD",
  "deliveryId": "<DELIVERYID>",
  "dropoffDeadline": "2023-06-02T17:33:50.000Z",
  "dropoffEta": "2023-06-02T17:00:07.000Z",
  "fee": 13.6,
  "manifestSize": 1,
  "pickupEta": "2023-06-02T16:18:14.000Z",
  "provider": "<PROVIDER>",
  "providerCreated": "2023-06-02T15:58:13.965Z",
  "providerInstance": "<PROVIDERINSTANCE>",
  "providerId": "<PROVIDERID>",
  "status": "PENDING",
  "statusHistory": [],
  "trackingURL": "https://<PROVIDER>.com/path/to/tracking/url"
}
```

Other Responses

In the event of the request not succeeding, a number of other responses may be returned.

Table 3-7 Other Responses — Request Not Successful

HTTP Code	Explanation	Body
400	The provided request input was not valid.	An array of MapperErrorData objects describing issues with the request.
404	A provided identifier or provider could not be found.	NA
424	One or more eligible providers were identified, but none were able to service the request.	An array of ProviderError objects explaining the issue(s).
500	The request failed for an unknown reason.	NA

Cancel

To cancel a delivery which has not yet reached a terminal state, the cancel endpoint is provided. This method cancels a single delivery using either its {id} or {deliveryId}.

POST

`https://<HOSTNAME>:<PORT>/<service>/api/delivery/v2/delivery/cancel?id={id}`

`https://<HOSTNAME>:<PORT>/<service>/api/delivery/v2/delivery/cancel?
deliveryId={deliveryId}`

Example Response

The following is an example of a successful response body in JSON format.

```
"created": "2023-06-02T15:58:14.260Z",
"createdBy": "<UID>",
"id": 1,
"barcodes": [
  {
    "created": "2023-06-02T15:58:14.298Z",
    "createdBy": "<UID>",
    "id": 1,
    "identifier": "<ID>",
    "parentId": <PARENTID>,
    "type": "QR",
    "value": "0000001 -01-01"
  }
],
"crossReferences": [],
"currency": "USD",
"deliveryId": "<DELIVERYID>",
"dropoffDeadline": "2023-06-02T17:33:50.000Z",
"dropoffEta": "2023-06-02T17:00:07.000Z",
"fee": 13.6,
"manifestSize": 1,
"pickupEta": "2023-06-02T16:18:14.000Z",
"provider": "<PROVIDER>",
"providerCreated": "2023-06-02T15:58:13.965Z",
"providerInstance": "<PROVIDERINSTANCE>",
"providerId": "<PROVIDERID>",
"status": "CANCELED",
"statusHistory": [],
"trackingURL": "https://<PROVIDER>.com/path/to/tracking/url"
}
```

Other Responses

In the event of the request not succeeding, a number of other responses may be returned.

Table 3-8 Other Responses — Request Not Successful

HTTP Code	Explanation	Body
400	The provider input is invalid	NA
404	A provided identifier or provider could not be found.	NA
409	Delivery is not in a status eligible for cancellation.	NA
424	One or more eligible providers were identified, but none were able to service the request.	An array of ProviderError objects explaining the issue(s).
500	The request failed for an unknown reason.	NA

Get

To retrieve a previously generated delivery, the get method is provided. This method retrieves a single delivery using its (id) identifier.

GET

`https://<HOSTNAME>:<PORT>/<service>/api/delivery/v2/delivery/{id}`

Example Response

The following is an example of a successful response body in JSON format.

```
{
  "created": "2023-06-02T15:58:14.260Z",
  "createdBy": "<UID>",
  "id": 1,
  "barcodes": [
    {
      "created": "2023-06-02T15:58:14.298Z",
      "createdBy": "<UID>",
      "id": 1,
      "identifier": "<ID>",
      "parentId": <PARENTID>,
      "type": "QR",
      "value": "0000001 -01-01"
    }
  ],
  "crossReferences": [],
  "currency": "USD",
  "deliveryId": "<DELIVERYID>",
  "dropoffDeadline": "2023-06-02T17:33:50.000Z",
  "dropoffEta": "2023-06-02T17:00:07.000Z",
  "fee": 13.6,
  "manifestSize": 1,
  "pickupEta": "2023-06-02T16:18:14.000Z",
  "provider": "<PROVIDER>",
```

```
"providerCreated": "2023-06-02T15:58:13.965Z",
"providerInstance": "<PROVIDERINSTANCE>",
"providerId": "<PROVIDERID>",
"status": "PENDING",
"statusHistory": [],
"trackingURL": "https://<PROVIDER>.com/path/to/tracking/url"
}
```

Other Responses

In the event of the request not succeeding, a number of other responses may be returned.

Table 3-9 Other Responses — Request Not Successful

HTTP Code	Explanation
404	The requested delivery could not be found.
500	The request failed for an unknown reason.

List

To obtain a list of deliveries, the list method is provided. This method returns an array of deliveries matching the provided criteria. See [Send Requests](#) for more information on this endpoint's available options.

GET

```
https://<HOSTNAME>:<PORT>/<service>/api/delivery/v2/delivery?
where={"deliveryId":["0000001"]}
```

Example Response

The following is an example of a successful response body in JSON format.

```
[
{
  "created": "2023-06-02T15:58:14.260Z",
  "createdBy": "<UID>",
  "id": 1,
  "barcodes": [
    {
      "created": "2023-06-02T15:58:14.298Z",
      "createdBy": "<UID>",
      "id": 1,
      "identifier": "<ID>",
      "parentId": <PARENTID>,
      "type": "QR",
      "value": "0000001 -01-01"
    }
  ],
  "crossReferences": [],
  "currency": "USD",
  "deliveryId": "<DELIVERYID>",
}
```

```
"dropoffDeadline": "2023-06-02T17:33:50.000Z",  
"dropoffEta": "2023-06-02T17:00:07.000Z",  
"fee": 13.6,  
"manifestSize": 1,  
"pickupEta": "2023-06-02T16:18:14.000Z",  
"provider": "<PROVIDER>",  
"providerCreated": "2023-06-02T15:58:13.965Z",  
"providerInstance": "<PROVIDERINSTANCE>",  
"providerId": "<PROVIDERID>",  
"status": "PENDING",  
"statusHistory": [],  
"trackingURL": "https://<PROVIDER>.com/path/to/tracking/url"  
}  
]
```

Other Responses

In the event of the request not succeeding, a number of other responses may be returned.

Table 3-10 Other Responses — Request Not Successful

HTTP Code	Explanation
500	The request failed for an unknown reason.

4

Collect and Receive Base APIs

This chapter describes the Collect and Receive base APIs:

- [Quote Services](#)
 - [Build a Quote — POST](#)
 - [Get a Quote — GET](#)
 - [Checks if a Quote Exists — HEAD](#)
 - [Queries for a Quote — GET](#)
 - [Queries for Quote Records/Count — GET](#)
 - [Queries for Quote Records by Attributes — GET](#)
- [Delivery Services](#)
 - [Build a Delivery — POST](#)
 - [Cancel a Delivery — POST](#)
 - [Get the Current Status of a Delivery — GET](#)
 - [Get a Delivery — GET](#)
 - [Checks if a Delivery Exists — HEAD](#)
 - [Queries for Delivery Records — GET](#)
 - [Queries for Delivery Records/Count — GET](#)
 - [Queries for Delivery Records by Attributes — GET](#)
- [Metrics Services](#)
 - [Metrics — GET](#)
 - [Location Metrics — GET](#)
- [Webhook Credential Services](#)
 - [Add Credential — POST](#)
 - [Get Credentials — GET](#)
 - [Get Credential — GET](#)
 - [Update Credential — PUT](#)
 - [Delete Credential — DELETE](#)
- [Status Notification Webhook Services](#)
 - [Add a New Webhook — POST](#)
 - [Queries for Webhook Records — GET](#)
 - [Performs a Bulk Action using a Collection of Webhook — POST](#)
 - [Queries for Webhook Records/Count — GET](#)
 - [Get a Webhook — GET](#)

- Update a Webhook — PUT
- Delete a Webhook — DELETE
- Checks If a Webhook Exists — HEAD
- Queries for Webhook Records By Attributes — GET
- Data Retention Configuration Service
 - Add a New DataRetentionConfig — POST
 - Queries for DataRetention Configuration Records — GET
 - Queries for DataRetention Configuration Records/Count — GET
 - Get a DataRetention Configuration — GET
 - Update a DataRetention Configuration — PUT
 - Delete a DataRetention Configuration — DELETE
 - Checks If a DataRetention Configuration Exists — HEAD
 - Queries for DataRetention Configuration Records By Attributes — GET

The Collect and Receive contains the following OpenAPI endpoint, which can be used to obtain the specification for the services it provides.

```
api/delivery/openapi
```

Quote Services

This section describes the Quote services.

Build a Quote — POST

This section describes the Build Quote endpoint. The Build Quote endpoint is used to check courier serviceability, validity, and cost for a delivery between the pickup and dropoff addresses. The Build Quote request gets a quote and should be made prior to submitting the request to Build a Delivery. The quote is valid for a short period of time and the quote ID returned in the response message is sent as part of the request to Build a Delivery.

Method

POST

URL

```
/api/delivery/v5/quote/build
```

Allowed Scope(s)

```
delivery-rw
```

Request

There are no request parameters for this operation.

The request content-type is application/json.

The ModelField request.

Example Value

```
{
  "deliveryType": "DELIVERY",
  "undeliverableAction": "LEAVE_AT_DOOR",
  "crossReferences": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "deliveryId": "string",
  "dropoffAddress": {
    "address1": "string",
    "address2": "string",
    "address3": "string",
    "address4": "string",
    "addressType": "RESIDENTIAL",
    "apartment": "string",
    "city": "string",
    "country": "string",
    "latitude": 0,
    "locationCode": "string",
    "longitude": 0,
    "postalCode": "string",
    "state": "string"
  },
  "dropoffContact": {
    "businessName": "string",
    "contactReference": "string",
    "emailAddress": "string",
    "firstName": "string",
    "lastName": "string",
    "middleName": "string",
    "phoneNumber": "string",
    "suffix": "string"
  },
  "dropoffDeadline": "2024-12-02T14:43:05.312Z",
  "dropoffNotes": "string",
  "dropoffOptions": {
    "additionalInstructions": "string",
    "barcodeRequired": false,
    "contactRequired": false,
    "minimumAge": 9999,
    "pictureRequired": false,
    "pincode": "string",
    "signatureRequired": false
  }
}
```

```
},
"dropoffReady": "2024-12-02T14:43:05.312Z",
"labelsAttached": false,
"manifest": [
  {
    "description": "string",
    "height": 9999,
    "identifier": "string",
    "items": [
      {
        "description": "string",
        "identifier": "string",
        "price": 10000000000000000,
        "quantity": 9999
      }
    ],
    "length": 9999,
    "packDimensionCode": "string",
    "totalAmount": 10000000000000000,
    "weight": 10000000000000000,
    "width": 9999
  }
],
"pickupAddress": {
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 0,
  "locationCode": "string",
  "longitude": 0,
  "postalCode": "string",
  "state": "string"
},
"pickupContact": {
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"pickupDeadline": "2024-12-02T14:43:05.312Z",
"pickupNotes": "string",
"pickupOptions": {
  "additionalInstructions": "string",
  "barcodeRequired": false,
  "contactRequired": false,
  "minimumAge": 9999,
  "pictureRequired": false,
```

```

    "pincode": "string",
    "signatureRequired": false
  },
  "pickupReady": "2024-12-02T14:43:05.312Z",
  "provider": "string",
  "quoteId": <quoteId>,
  "tip": 10000000000000000,
  "totalValue": 10000000000000000
}

```

Schema — Delivery Request

The following table describes details of proposed delivery request.

Table 4-1 Delivery Request — Object

Element Name	Required	Data Type	Description
deliveryType	NA	string	The delivery type identifier. Default: DELIVERY; Enum: DELIVERY; RETURN
undeliverableAction	NA	string	Indicates what the courier should do with the packages in the event they are unable to be delivered. Enum: LEAVE_AT_DOOR, RETURN
crossReferences	NA	object	Cross references for the delivery. Intended for use by the retailer as an additional reference stored with the delivery order.
deliveryId	Yes	string (128)	The primary reference ID from the calling system uniquely identifying this delivery. Multiple quotes may be obtained, but only one delivery may be arranged for a given reference ID.
dropoffAddress	Yes	object	An address for a given dropoff location.
dropoffContact	Yes	object	A point of contact for a given dropoff location.
dropoffDeadline	NA	string <date-time>	The time by which the items must be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.
dropoffNotes	NA	string	Notes for the courier at drop-off location.
dropoffOptions	NA	object	Options for customizing required verification.
dropoffReady	NA	string <date-time>	The time at which the items may be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.

Table 4-1 (Cont.) Delivery Request — Object

Element Name	Required	Data Type	Description
labelsAttached	NA	boolean	Indicates whether the package(s) will have a pre-paid shipping label attached when the courier arrives. Default: false Note: Return delivery types have the following limitations with the Uber integration: Labels Attached will always be set to TRUE which indicates that a label will be attached to the package(s).
manifest	Yes	object	The packages to be delivered.
pickupAddress	Yes	object	An address given pickup location.
pickupContact	Yes	object	A point of contact for a given pickup location.
pickupDeadline	NA	string <date-time>	The time by which the items must be picked up. Date-time strings are in ISO 8601 timestamp with offset format.
pickupNotes	NA	string	Notes for the courier at pickup location.
pickupOptions	NA	object	Options for customizing required verification.
pickupReady	NA	string <date-time>	The time at which the items will be ready for pickup. Date-time strings are in ISO 8601 timestamp with offset format.
provider	NA	string (10)	The preferred provider.
quoteId	NA	integer <int64>	The prior quote ID for the delivery, if applicable.
tip	NA	number	The amount of the tip.
totalValue	Yes	number	The total value of the request.

The following table describes the elements of an address for a pickup or delivery.

Table 4-2 Request Address

Element Name	Required	Data Type	Description
address1	Yes	string (254)	The address line 1.
address2	NA	string (254)	The address line 2.
address3	NA	string (254)	The address line 3.
address4	NA	string (254)	The address line 4.
addressType	NA	string	enum: RESIDENTIAL, BUSINESS
apartment	NA	string (254)	The apartment.
city	Yes	string (254)	The city or town.
country	Yes	string (254)	The ISO 3166 country code. Applications can decide whether to use 2 or 3 char codes.

Table 4-2 (Cont.) Request Address

Element Name	Required	Data Type	Description
latitude	NA	number	The latitude of the address.
locationCode	NA	string (10)	The location code for the address, if applicable. This must be unique across the retailer's landscape.
longitude	NA	number	The longitude of the address.
postalCode	NA	string (20)	The postal code.
state	NA	string (254)	The ISO 3166 state, province or territory code.

The following table describes the elements of a point of contact for a given pickup or delivery.

Table 4-3 Request Contact

Element Name	Required	Data Type	Description
businessName	NA	string (254)	The business name.
contactReference	NA	string (64)	A reference number for the contact.
emailAddress	NA	string (254)	The email address.
firstName	NA	string (254)	The first (that is, given) name.
lastName	NA	string (254)	The last (that is, family) name.
middleName	NA	string (254)	The middle name.
phoneNumber	Yes	string (50)	The phone number.
suffix	NA	string (254)	A name suffix.

The following table describes the elements of options for customizing required verification.

Table 4-4 Verification Options

Element Name	Required	Data Type	Description
additionalInstructions	NA	string	Additional instructions for the courier at handoff.
barcodeRequired	NA	boolean	Indicates that a barcode scan is required. Default: false
contactRequired	NA	boolean	Indicates that contact is required (that is must NOT be contactless). Default: false
minimumAge	NA	integer <int32>	The minimum age of the recipient. If this value is specified, Uber will reject the request if the age is not between 18 and 25. Note: Return delivery types have the following limitations with the Uber integration: The use of Minimum Age is not supported as drop off verification option.

Table 4-4 (Cont.) Verification Options

Element Name	Required	Data Type	Description
pictureRequired	NA	boolean	Indicates that a picture is required. Default: false Note: Return delivery types have the following limitations with the Uber integration: Picture Required will always be set to TRUE which requires the courier to take a picture as a drop off verification.
pincode	NA	string	A pincode required for entry at handoff. Note: Return delivery types have the following limitations with the Uber integration: The use of Pincodes are not supported as drop off verification option.
signatureRequired	NA	boolean	Indicates that a signature is required. Default: false

The following table describes the elements of a package of items to be delivered.

Table 4-5 Manifest Package — Object

Element Name	Required	Data Type	Description
description	NA	string (128)	The description of the package.
height	Yes	integer <int32>	The height of the item in centimeters.
identifier	Yes	string (100)	A unique identifier for the package. This value will be considered when producing verification barcodes.
items	NA	object	The items to be delivered.
length	Yes	integer <int32>	The length of the item in centimeters.
packDimensionCode	NA	string	The package dimension code.
totalAmount	NA	number	The combined prices of all items within the package.
weight	Yes	number	The weight of the item in kilograms.
width	Yes	integer <int32>	The width of the item in centimeters.

The following table describes the elements of an individual item within a delivery.

Table 4-6 Item — Object

Element Name	Required	Data Type	Description
description	Yes	string (128)	The description of the item.
identifier	Yes	string (100)	The identifier (UPC, EAN, SKU, and so on) of the item.
price	NA	number	The price of the item.
quantity	Yes	integer <int32>	The quantity of the item.

Responses

This section describes the responses of the buildQuote endpoint.

Response Code: 200

The result object.

The media type is application/json.

Example Value

```
{
  "errors": [
    {
      "errorCode": 2147483647,
      "errorMessage": "string",
      "moreInfo": {
        "code": "string",
        "description": "string"
      },
      "provider": "string"
    }
  ],
  "quotes": [
    {
      "created": "2024-05-23T13:44:28.179Z",
      "createdBy": "string",
      "updated": "2024-05-23T13:44:28.179Z",
      "updatedBy": "string",
      "id": <id>,
      "crossReferences": [
        {
          "created": "2024-05-23T13:44:28.179Z",
          "createdBy": "string",
          "updated": "2024-05-23T13:44:28.179Z",
          "updatedBy": "string",
          "id": <id>,
          "key": "string",
          "parentId": <parentId>,
          "value": "string"
        }
      ],
      "currency": "str",
      "deliveryId": "string",
      "deliveryType": "DELIVERY",
      "dropoffAddress": {
        "created": "2024-05-23T13:44:28.179Z",
        "createdBy": "string",
        "updated": "2024-05-23T13:44:28.179Z",
        "updatedBy": "string",
        "id": <id>,
        "address1": "string",
```

```
"address2": "string",
"address3": "string",
"address4": "string",
"addressType": "RESIDENTIAL",
"apartment": "string",
"city": "string",
"country": "string",
"latitude": 999999999,
"locationCode": "string",
"longitude": 999999999,
"postalCode": "string",
"state": "string"
},
"dropoffAddressId": <dropoffAddressId>,
"dropoffContact": {
  "created": "2024-05-23T13:44:28.179Z",
  "createdBy": "string",
  "updated": "2024-05-23T13:44:28.179Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"dropoffContactId": <dropoffContactId>,
"dropoffDeadline": "2024-05-23T13:44:28.179Z",
"dropoffEta": "2024-05-23T13:44:28.179Z",
"dropoffLocationCode": "string",
"expires": "2024-05-23T13:44:28.179Z",
"fee": 10000000000000000,
"pickupAddress": {
  "created": "2024-05-23T13:44:28.179Z",
  "createdBy": "string",
  "updated": "2024-05-23T13:44:28.179Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"pickupAddressId": <pickupAddressId>
```

```
    "pickupContact": {
      "created": "2024-05-23T13:44:28.179Z",
      "createdBy": "string",
      "updated": "2024-05-23T13:44:28.179Z",
      "updatedBy": "string",
      "id": <id>,
      "businessName": "string",
      "contactReference": "string",
      "emailAddress": "string",
      "firstName": "string",
      "lastName": "string",
      "middleName": "string",
      "phoneNumber": "string",
      "suffix": "string"
    },
    "pickupContactId": <pickupContactId>,
    "pickupDeadline": "2024-05-23T13:44:28.179Z",
    "pickupEta": "2024-05-23T13:44:28.179Z",
    "pickupLocationCode": "string",
    "provider": "string",
    "providerCreated": "2024-05-23T13:44:28.179Z",
    "providerId": "string",
    "providerInstance": "string"
  }
}
```

Schema — Quote Response

The following table describes the elements of a response to a quote request.

Table 4-7 Quote Response — Object

Element Name	Required	Data Type	Description
errors	NA	object	A collection of errors from providers which could not service the request.
quotes	NA	object	A collection of quotes for the provided request.

The following table describes elements of a container for an error from a provider.

Table 4-8 Provider Errors — Object

Element Name	Required	Data Type	Description
errorCode	NA	integer <int32>	A numeric error indicating the type of error.
errorMessage	NA	string	An unlocalized message indicating details about the error.
moreInfo	NA	object	An object containing provider-specific details pertaining to an error.
provider	Yes	string (10)	The provider associated with the error.

The following table describes elements of an object containing provider-specific details pertaining to an error.

Table 4-9 More Error Information — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

The following table describes elements of a quote from a delivery provider for a proposed delivery.

Table 4-10 Quote — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
crossReferences	NA	Collection of [object]	A list of cross-references for the quote.
currency	NA	string (3)	The currency code associated with the quote.
deliveryId	Yes	string (128)	The primary reference ID from the calling system uniquely identifying the delivery associated with this quote.
deliveryType	Yes	string	The delivery type identifier. Default: DELIVERY; Enum: DELIVERY, RETURN
dropoffAddress	NA	object	An address for a given dropoff location.
dropoffAddressId	NA	integer <int64>	Drop Off Address ID
dropoffContact	NA	object	A point of contact for a given dropoff location.
dropoffContactId	NA	integer <int64>	Drop Off Contact ID
dropoffDeadline	NA	string <date-time>	The time by which the order must be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.
dropoffEta	NA	string <date-time>	The time by which the order is expected to be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.

Table 4-10 (Cont.) Quote — Object

Element Name	Required	Data Type	Description
dropoffLocationCode	NA	string(10)	The location code for the dropoff address.
expires	NA	string <date-time>	The time at which this quote will no longer be accepted. Date-time strings are in ISO 8601 timestamp with offset format.
fee	NA	number	The cost of the delivery
pickupAddress	NA	object	An address for a given pickup location.
pickupAddressId	NA	integer <int64>	Pick Up Address ID.
pickupContact	NA	object	A point of contact for a given pickup location.
pickupContactId	NA	integer <int64>	Pick Up Contact ID.
pickupDeadline	NA	string <date-time>	The time by which the order must be picked up. Date-time strings are in ISO 8601 timestamp with offset format.
pickupEta	NA	string <date-time>	The time by which the order is expected to be picked up. Date-time strings are in ISO 8601 timestamp with offset format.
pickupLocationCode	NA	string (10)	The pickup location code associated with this quote.
provider	Yes	string (10)	The provider of the quote.
providerCreated	Yes	string <date-time>	The creation timestamp, as indicated by the provider. Date-time strings are in ISO 8601 timestamp with offset format.
providerInstance	Yes	string (20)	The specific instance of the provider of the quote.
providerId	Yes	string (128)	The id from the provider of the quote.

The following table describes the elements of an address for a pickup or delivery.

Table 4-11 Address

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.

Table 4-11 (Cont.) Address

Element Name	Required	Data Type	Description
id	NA	<int64>	ID
address1	Yes	string (254)	The address line 1.
address2	NA	string (254)	The address line 2.
address3	NA	string (254)	The address line 3.
address4	NA	string (254)	The address line 4.
addressType	NA	string	enum: RESIDENTIAL, BUSINESS
apartment	NA	string (254)	The apartment.
city	Yes	string (254)	The city or town.
country	Yes	string (254)	The ISO 3166 country code. Applications can decide whether to use 2 or 3 char codes.
latitude	NA	number	The latitude of the address.
locationCode	NA	string (10)	The location code for the address, if applicable. This must be unique across the retailer's landscape.
longitude	NA	number	The longitude of the address.
postalCode	NA	string (20)	The postal code.
state	NA	string (254)	The ISO 3166 state, province or territory code.

The following table describes the elements of a point of contact for a given pickup or delivery.

Table 4-12 Contact

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who last created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	<int64>	ID
businessName	NA	string (254)	The business name.
contactReference	NA	string (64)	A reference number for the contact.
emailAddress	NA	string (254)	The email address.
firstName	NA	string (254)	The first (that is, given) name.
lastName	NA	string (254)	The last (that is, family) name.
middleName	NA	string (254)	The middle name.
phoneNumber	Yes	string (50)	The phone number.

Table 4-12 (Cont.) Contact

Element Name	Required	Data Type	Description
suffix	NA	string (254)	A name suffix.

The following table describes the elements of a cross reference record.

Table 4-13 Quote Cross References — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
key	Yes	string (128)	The identifier for the cross reference.
parentId	NA	integer <int64>	Parent ID
value	Yes	string (128)	The the value of the cross reference.

Response Code: 204

No providers were eligible to service this request.

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
```

```
    "errorType": "INTEGRATION"  
  }  
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 4-14 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 4-15 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Response Code: 404

Provided quote ID or associated provider could not be found.

Response Code: 424

No providers could service this request.

The media type is application/json.

Example Value

```
[
  {
    "errorCode": 2147483647,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "provider": "string"
  }
]
```

Schema — Provider Error

The following table describes elements of a container for an error from a provider.

Table 4-16 Provider Error — Object

Element Name	Required	Data Type	Description
errorCode	NA	integer <int32>	A numeric error indicating the type of error.
errorMessage	NA	string	An unlocalized message indicating details about the error.
moreInfo	NA	object	An object containing provider-specific details pertaining to an error.
provider	Yes	string (10)	The provider associated with the error.

The following table describes elements of an object containing provider-specific details pertaining to an error.

Table 4-17 More Error Information — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Response Code: 500

The request failed.

Get a Quote — GET

This section describes the Get a Quote endpoint. Use the Get a Quote endpoint to get the quote details for an existing quote record.

Method

GET

URL

/api/delivery/v5/quote/{id}

Allowed Scope(s)

delivery-r
delivery-rw

Request

This section describes the request parameters.

Table 4-18 Get a Quote — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer <int64>	ID
expands	NA	string	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Responses

This section describes the Get a Quote endpoint responses.

Response Code: 200

The requested record.
The media type is application/json.

Example Value

```
{  
  "created": "2024-05-23T14:17:10.839Z",
```

```
"createdBy": "string",
"updated": "2024-05-23T14:17:10.839Z",
"updatedBy": "string",
"id": <id>,
"crossReferences": [
  {
    "created": "2024-05-23T14:17:10.839Z",
    "createdBy": "string",
    "updated": "2024-05-23T14:17:10.839Z",
    "updatedBy": "string",
    "id": <id>,
    "key": "string",
    "parentId": 9223372036854776000,
    "value": "string"
  }
],
"currency": "str",
"deliveryId": "string",
"deliveryType": "DELIVERY",
"dropoffAddress": {
  "created": "2024-05-23T14:17:10.839Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:17:10.839Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"dropoffAddressId": <dropoffAddressId>,
"dropoffContact": {
  "created": "2024-05-23T14:17:10.839Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:17:10.839Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"dropoffContactId": <dropoffContactId>
```

```

"dropoffDeadline": "2024-05-23T14:17:10.839Z",
"dropoffEta": "2024-05-23T14:17:10.839Z",
"dropoffLocationCode": "string",
"expires": "2024-05-23T14:17:10.839Z",
"fee": 10000000000000000,
"pickupAddress": {
  "created": "2024-05-23T14:17:10.839Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:17:10.839Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"pickupAddressId": <pickupAddressId>,
"pickupContact": {
  "created": "2024-05-23T14:17:10.839Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:17:10.839Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"pickupContactId": <pickupContactId>,
"pickupDeadline": "2024-05-23T14:17:10.839Z",
"pickupEta": "2024-05-23T14:17:10.839Z",
"pickupLocationCode": "string",
"provider": "string",
"providerCreated": "2024-05-23T14:17:10.839Z",
"providerId": "string",
"providerInstance": "string"
}

```

Schema — Quote

The tables below describe the elements of a quote from a delivery provider for a proposed delivery.

Table 4-19 Quote — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
crossReferences	NA	object	A list of cross-references for the quote.
currency	NA	string (3)	The currency code associated with the quote.
deliveryId	Yes	string (128)	The primary reference ID from the calling system uniquely identifying the delivery associated with this quote.
deliveryType	Yes	string	The delivery type identifier. Default: DELIVERY; Enum: DELIVERY, RETURN
dropoffAddress	NA	object	An address for a given dropoff location.
dropoffAddressId	NA	integer <int64>	NA
dropoffContact	NA	object	A point of contact for a given dropoff location.
dropoffContactId	NA	integer <int64>	NA
dropoffDeadline	NA	string <date-time>	The time by which the order must be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.
dropoffEta	NA	string <date-time>	The time by which the order is expected to be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.
dropoffLocationCode	NA	string(10)	The location code for the dropoff address.
expires	NA	string <date-time>	The time at which this quote will no longer be accepted. Date-time strings are in ISO 8601 timestamp with offset format.
fee	NA	number	The cost of the delivery.
pickupAddress	Yes	object	An address given pickup location.
pickupAddressId	NA	integer <int64>	NA
pickupContact	NA	object	A point of contact for a given pickup location.

Table 4-19 (Cont.) Quote — Object

Element Name	Required	Data Type	Description
pickupContactId	NA	integer <int64>	NA
pickupDeadline	NA	string <date-time>	The time by which the order must be picked up. Date-time strings are in ISO 8601 timestamp with offset format.
pickupEta	NA	string <date-time>	The time by which the order is expected to be picked up. Date-time strings are in ISO 8601 timestamp with offset format.
pickupLocationCode	NA	string (10)	The pickup location code associated with this quote.
provider	Yes	string (10)	The provider of the quote.
providerCreated	Yes	string <date-time>	The creation timestamp, as indicated by the provider. Date-time strings are in ISO 8601 timestamp with offset format.
providerInstance	Yes	string (20)	The specific instance of the provider of the quote.
providerId	Yes	string (128)	The ID from the provider of the quote.

The following table describes the elements of a cross reference record for a quote from a delivery provider for a proposed delivery.

Table 4-20 Quote Cross Reference — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
key	Yes	string (128)	The identifier for the cross reference.
parentId	NA	integer <int64>	Parent ID
value	Yes	string (128)	The the value of the cross reference.

The following table describes the elements of an address for a given pickup or delivery.

Table 4-21 Address

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
address1	Yes	string (254)	The address line 1.
address2	NA	string (254)	The address line 2.
address3	NA	string (254)	The address line 3.
address4	NA	string (254)	The address line 4.
addressType	NA	string	enum: RESIDENTIAL, BUSINESS
apartment	NA	string (254)	The apartment.
city	Yes	string (254)	The city or town.
country	Yes	string (254)	The ISO 3166 country code. Applications can decide whether to use 2 or 3 char codes.
latitude	NA	number	The latitude of the address.
locationCode	NA	string (10)	The location code for the address, if applicable. This must be unique across the retailer's landscape.
longitude	NA	number	The longitude of the address.
postalCode	NA	string (20)	The postal code.
state	NA	string (254)	The ISO 3166 state, province or territory code.

The following table describes the elements of a point of contact for a given pickup or delivery.

Table 4-22 Contact

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.

Table 4-22 (Cont.) Contact

Element Name	Required	Data Type	Description
id	NA	integer <int64>	ID
businessName	NA	string (254)	The business name.
contactReference	NA	string (64)	A reference number for the contact.
emailAddress	NA	string (254)	The email address.
firstName	NA	string (254)	The first (that is, given) name.
lastName	NA	string (254)	The last (that is, family) name.
middleName	NA	string (254)	The middle name.
phoneNumber	Yes	string (50)	The phone number.
suffix	NA	string (254)	A name suffix.

Response Code: 404

The record could not be found.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Checks if a Quote Exists — HEAD

This section describes the endpoint which checks if a quote exists.

Method

HEAD

URL

/api/delivery/v5/quote/{id}

Allowed Scope(s)

delivery-r
delivery-rw

Request

This section describes the request parameters.

Table 4-23 Checks if a Quote Exists — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer <int64> (path)	ID

Responses

This section describes the responses of the Checks if a Quote Exists endpoint.

Response Code: 200

The requested record exists.

Response Code: 404

The record could not be found.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for a Quote — GET

This section describes the Queries for a Quote endpoint. This service returns a list of quote records and the associated quote details matching the provided criteria.

Method

GET

URL

/api/delivery/v5/quote

Allowed Scope(s)

delivery-r
delivery-rw

Request

This section describes the request parameters.

Table 4-24 Queries for a Quote Records— Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, that is a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example, {"child.name":"Jo"} would match Joe, John, and so on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example, {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example, {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is "@and1",</p>

Table 4-24 (Cont.) Queries for a Quote Records— Request Parameters

Parameter	Required	Data Type	Description
orderBy	NA	string (query)	""@and2"", and so on) to ensure that they are uniquely named since any duplicates will be eliminated. A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by " asc " (default) or " desc " to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.
limit	NA	integer <int32> (query)	Provides a maximum number of records to be returned.
offset	NA	integer <int32> (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the Queries for a Quote endpoint responses.

Response Code: 200

The requested record.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-05-23T14:41:45.940Z",
    "createdBy": "string",
    "updated": "2024-05-23T14:41:45.940Z",
```

```
"updatedAt": "string",
"id": <id>,
"crossReferences": [
  {
    "created": "2024-05-23T14:41:45.940Z",
    "createdBy": "string",
    "updated": "2024-05-23T14:41:45.940Z",
    "updatedAt": "string",
    "id": <id>,
    "key": "string",
    "parentId": <parentId>,
    "value": "string"
  }
],
"currency": "str",
"deliveryId": "string",
"deliveryType": "DELIVERY",
"dropoffAddress": {
  "created": "2024-05-23T14:41:45.940Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:41:45.940Z",
  "updatedAt": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"dropoffAddressId": <dropoffAddressId>,
"dropoffContact": {
  "created": "2024-05-23T14:41:45.940Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:41:45.940Z",
  "updatedAt": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"dropoffContactId": <dropoffContactId>,
"dropoffDeadline": "2024-05-23T14:41:45.940Z",
"dropoffEta": "2024-05-23T14:41:45.940Z",
```

```

"dropoffLocationCode": "string",
"expires": "2024-05-23T14:41:45.940Z",
"fee": 10000000000000000,
"pickupAddress": {
  "created": "2024-05-23T14:41:45.940Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:41:45.940Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"pickupAddressId": <pickupAddressId>,
"pickupContact": {
  "created": "2024-05-23T14:41:45.940Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:41:45.940Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"pickupContactId": <pickupContactId>,
"pickupDeadline": "2024-05-23T14:41:45.940Z",
"pickupEta": "2024-05-23T14:41:45.940Z",
"pickupLocationCode": "string",
"provider": "string",
"providerCreated": "2024-05-23T14:41:45.940Z",
"providerId": "string",
"providerInstance": "string"
}
]

```

Schema — Quote

For more information about the Quote schema, see the [Schema — Quote](#) section.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Quote Records/Count— GET

This section describes the Queries for Quote Records Count endpoint. This service returns a count of records matching the provided criteria.

Method

GET

URL

`/api/delivery/v5/quote/count`

Allowed Scope(s)

`delivery-r`
`delivery-rw`

Request

This section describes the request parameters.

Table 4-25 Queries for Quote Records Count — Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, i.e. a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example, {"child.name":"Jo"} would match Joe, John, and so on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example, {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example, {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is "@and1",</p>

Table 4-25 (Cont.) Queries for Quote Records Count — Request Parameters

Parameter	Required	Data Type	Description
			"@and2", and so on) to ensure that they are uniquely named since any duplicates will be eliminated.

Responses

This section describes the responses of the Queries for Quote Records Count endpoint.

Response Code: 200

The count of the requested records.

The media type is application/json.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Quote Records by Attributes — GET

This section describes the Queries for Quote Records by Attributes endpoint. This endpoint returns a list of records matching the provided attributes.

Method

GET

URL

/api/delivery/v5/quote/byattributes

Allowed Scope(s)

delivery-r

Request

This section describes the request parameters.

Table 4-26 Queries for Quote Records by Attributes — Request Parameters

Parameter	Required	Data Type	Description
id	NA	integer (\$int64) (query)	id
currency	NA	string (query)	currency
deliveryId	NA	string (query)	delivery id
deliveryType	NA	string (query)	delivery type; Available values : DELIVERY, RETURN; Default value: DELIVERY
dropoffAddressId	NA	integer (\$int64) (query)	dropoff address ID
dropoffContactId	NA	integer (\$int64) (query)	dropoff contact ID
dropoffDeadline	NA	string (\$date-time) (query)	dropoff deadline
dropoffEta	NA	string (\$date-time) (query)	dropoff Eta
dropoffLocationCode		string (query)	dropoff location code
expires	NA	string (\$date-time) (query)	expires
fee	NA	number (query)	fee
pickupAddressId	NA	integer (\$int64) (query)	pickup address ID
pickupContactId	NA	integer (\$int64) (query)	pickup contact ID
pickupDeadline	NA	string (\$date-time) (query)	pickup deadline
pickupEta	NA	string (\$date-time) (query)	pickup Eta
pickupLocationCode	NA	string (query)	pickup location code
provider	NA	string (query)	provider
providerCreated	NA	string (\$date-time) (query)	providerCreated
providerId	NA	string (query)	provider ID
providerInstance	NA	string (query)	provider instance
orderBy	NA	string (query)	A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by "asc" (default) or "desc" to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.

Table 4-26 (Cont.) Queries for Quote Records by Attributes — Request Parameters

Parameter	Required	Data Type	Description
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.
limit	NA	integer (\$int32) (query)	Provides a maximum number of records to be returned.
offset	NA	integer (\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for Quote Records by Attributes endpoint.

Response Code: 200

The requested record.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-05-23T14:47:57.555Z",
    "createdBy": "string",
    "updated": "2024-05-23T14:47:57.555Z",
    "updatedBy": "string",
    "id": <id>,
    "crossReferences": [
      {
        "created": "2024-05-23T14:47:57.555Z",
        "createdBy": "string",
        "updated": "2024-05-23T14:47:57.555Z",
        "updatedBy": "string",
        "id": <id>,
        "key": "string",
        "parentId": <parentId>,
        "value": "string"
      }
    ],
    "currency": "str",
```

```
"deliveryId": "string",
"deliveryType": "DELIVERY",
"dropoffAddress": {
  "created": "2024-05-23T14:47:57.555Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:47:57.555Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"dropoffAddressId": 9223372036854776000,
"dropoffContact": {
  "created": "2024-05-23T14:47:57.555Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:47:57.555Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"dropoffContactId": <dropoffContactId>,
"dropoffDeadline": "2024-05-23T14:47:57.555Z",
"dropoffEta": "2024-05-23T14:47:57.555Z",
"dropoffLocationCode": "string",
"expires": "2024-05-23T14:47:57.555Z",
"fee": 10000000000000000,
"pickupAddress": {
  "created": "2024-05-23T14:47:57.555Z",
  "createdBy": "string",
  "updated": "2024-05-23T14:47:57.555Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
```

```

        "city": "string",
        "country": "string",
        "latitude": 999999999,
        "locationCode": "string",
        "longitude": 999999999,
        "postalCode": "string",
        "state": "string"
    },
    "pickupAddressId": <pickupAddressId>,
    "pickupContact": {
        "created": "2024-05-23T14:47:57.555Z",
        "createdBy": "string",
        "updated": "2024-05-23T14:47:57.555Z",
        "updatedBy": "string",
        "id": <id>,
        "businessName": "string",
        "contactReference": "string",
        "emailAddress": "string",
        "firstName": "string",
        "lastName": "string",
        "middleName": "string",
        "phoneNumber": "string",
        "suffix": "string"
    },
    "pickupContactId": <pickupContactId>,
    "pickupDeadline": "2024-05-23T14:47:57.555Z",
    "pickupEta": "2024-05-23T14:47:57.555Z",
    "pickupLocationCode": "string",
    "provider": "string",
    "providerCreated": "2024-05-23T14:47:57.555Z",
    "providerId": "string",
    "providerInstance": "string"
    }
}
]

```

Schema — Quote

For more information about the Quote schema, see the [Schema — Quote](#) section.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Delivery Services

This section describes the Delivery services.

Build a Delivery — POST

This section describes the Build Delivery endpoint. Use the Build Delivery endpoint to create a delivery with the courier provider. This includes but is not limited to: the pickup and dropoff information, package details, scheduling and delivery verification options.

Method

POST

URL

/api/delivery/v5/delivery/build

Allowed Scope(s)

delivery-rw

Request

There are no request parameters for this operation.

The request body is application/json.

The ModelField request.

Example Value

```
{
  "deliveryType": "DELIVERY",
  "undeliverableAction": "LEAVE_AT_DOOR",
  "crossReferences": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "deliveryId": "string",
  "dropoffAddress": {
    "address1": "string",
    "address2": "string",
    "address3": "string",
    "address4": "string",
    "addressType": "RESIDENTIAL",
    "apartment": "string",
    "city": "string",
    "country": "string",
    "latitude": 0,
    "locationCode": "string",
    "longitude": 0,
    "postalCode": "string",
    "state": "string"
  },
  "dropoffContact": {
    "businessName": "string",
    "contactReference": "string",
```

```
"emailAddress": "string",
"firstName": "string",
"lastName": "string",
"middleName": "string",
"phoneNumber": "string",
"suffix": "string"
},
"dropoffDeadline": "2024-11-28T12:22:24.965Z",
"dropoffNotes": "string",
"dropoffOptions": {
  "additionalInstructions": "string",
  "barcodeRequired": false,
  "contactRequired": false,
  "minimumAge": 9999,
  "pictureRequired": false,
  "pincode": "string",
  "signatureRequired": false
},
"dropoffReady": "2024-11-28T12:22:24.965Z",
"labelsAttached": false,
"manifest": [
  {
    "description": "string",
    "height": 9999,
    "identifier": "string",
    "items": [
      {
        "description": "string",
        "identifier": "string",
        "price": 10000000000000000,
        "quantity": 9999
      }
    ],
    "length": 9999,
    "packDimensionCode": "string",
    "totalAmount": 10000000000000000,
    "weight": 10000000000000000,
    "width": 9999
  }
],
"pickupAddress": {
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 0,
  "locationCode": "string",
  "longitude": 0,
  "postalCode": "string",
  "state": "string"
},
"pickupContact": {
```

```
    "businessName": "string",
    "contactReference": "string",
    "emailAddress": "string",
    "firstName": "string",
    "lastName": "string",
    "middleName": "string",
    "phoneNumber": "string",
    "suffix": "string"
  },
  "pickupDeadline": "2024-11-28T12:22:24.965Z",
  "pickupNotes": "string",
  "pickupOptions": {
    "additionalInstructions": "string",
    "barcodeRequired": false,
    "contactRequired": false,
    "minimumAge": 9999,
    "pictureRequired": false,
    "pincode": "string",
    "signatureRequired": false
  },
  "pickupReady": "2024-11-28T12:22:24.965Z",
  "provider": "string",
  "quoteId": 9223372036854776000,
  "tip": 10000000000000000,
  "totalValue": 10000000000000000
}
```

Schema — Delivery Request

The following table describes details of proposed delivery request.

Table 4-27 Delivery Request — Object

Element Name	Required	Data Type	Description
deliveryType	NA	string	The delivery type identifier. Default: DELIVERY; Enum: DELIVERY; RETURN
undeliverableAction	NA	string	Indicates what the courier should do with the packages in the event they are unable to be delivered. Enum: LEAVE_AT_DOOR, RETURN
crossReferences	NA	object	Cross references for the delivery. Intended for use by the retailer as an additional reference stored with the delivery order.
deliveryId	Yes	string (128)	The primary reference ID from the calling system uniquely identifying this delivery. Multiple quotes may be obtained, but only one delivery may be arranged for a given reference ID.
dropoffAddress	Yes	object	An address for a given dropoff location.

Table 4-27 (Cont.) Delivery Request — Object

Element Name	Required	Data Type	Description
dropoffContact	Yes	object	A point of contact for a given dropoff location.
dropoffDeadline	NA	string <date-time>	The time by which the items must be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.
dropoffNotes	NA	string	Notes for the courier at drop-off location.
dropoffOptions	NA	object	Options for customizing required verification.
dropoffReady	NA	string <date-time>	The time at which the items may be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.
labelsAttached	NA	boolean	Indicates whether the package(s) will have a pre-paid shipping label attached when the courier arrives. Default: false Note: Return delivery types have the following limitations with the Uber integration: Labels Attached will always be set to TRUE which indicates that a label will be attached to the package(s).
manifest	Yes	object	The packages to be delivered.
pickupAddress	Yes	object	An address given pickup location.
pickupContact	Yes	object	A point of contact for a given pickup location.
pickupDeadline	NA	string <date-time>	The time by which the items must be picked up. Date-time strings are in ISO 8601 timestamp with offset format.
pickupNotes	NA	string	Notes for the courier at pickup location.
pickupOptions	NA	object	Options for customizing required verification.
pickupReady	NA	string <date-time>	The time at which the items will be ready for pickup. Date-time strings are in ISO 8601 timestamp with offset format.
provider	NA	string (10)	The preferred provider.
quoteId	NA	integer <int64>	The prior quote ID for the delivery, if applicable.
tip	NA	number	The amount of the tip.
totalValue	Yes	number	The total value of the request.

The following table describes the elements of an address for a pickup or delivery.

Table 4-28 Request Address

Element Name	Required	Data Type	Description
address1	Yes	string (254)	The address line 1.
address2	NA	string (254)	The address line 2.
address3	NA	string (254)	The address line 3.
address4	NA	string (254)	The address line 4.
addressType	NA	string	enum: RESIDENTIAL, BUSINESS
apartment	NA	string (254)	The apartment.
city	Yes	string (254)	The city or town.
country	Yes	string (254)	The ISO 3166 country code. Applications can decide whether to use 2 or 3 char codes.
latitude	NA	number	The latitude of the address.
locationCode	NA	string (10)	The location code for the address, if applicable. This must be unique across the retailer's landscape.
longitude	NA	number	The longitude of the address.
postalCode	NA	string (20)	The postal code.
state	NA	string (254)	The ISO 3166 state, province or territory code.

The following table describes the elements of a point of contact for a given pickup or delivery.

Table 4-29 Request Contact

Element Name	Required	Data Type	Description
businessName	NA	string (254)	The business name.
contactReference	NA	string (64)	A reference number for the contact.
emailAddress	NA	string (254)	The email address.
firstName	NA	string (254)	The first (that is, given) name.
lastName	NA	string (254)	The last (that is, family) name.
middleName	NA	string (254)	The middle name.
phoneNumber	Yes	string (50)	The phone number.
suffix	NA	string (254)	A name suffix.

The following table describes the elements of options for customizing required verification.

Table 4-30 Verification Options

Element Name	Required	Data Type	Description
additionalInstructions	NA	string	Additional instructions for the courier at handoff.
barcodeRequired	NA	boolean	Indicates that a barcode scan is required. Default: false

Table 4-30 (Cont.) Verification Options

Element Name	Required	Data Type	Description
contactRequired	NA	boolean	Indicates that contact is required (that is must NOT be contactless). Default: false
minimumAge	NA	integer <int32>	The minimum age of the recipient. If this value is specified, Uber will reject the request if the age is not between 18 and 25. Note: Return delivery types have the following limitations with the Uber integration: The use of Minimum Age is not supported as drop off verification option.
pictureRequired	NA	boolean	Indicates that a picture is required. Default: false Note: Return delivery types have the following limitations with the Uber integration: Picture Required will always be set to TRUE which requires the courier to take a picture as a drop off verification.
pincode	NA	string	A pincode required for entry at handoff. Note: Return delivery types have the following limitations with the Uber integration: The use of Pincodes are not supported as drop off verification option.
signatureRequired	NA	boolean	Indicates that a signature is required. Default: false

The following table describes the elements of a package of items to be delivered.

Table 4-31 Manifest Package — Object

Element Name	Required	Data Type	Description
description	NA	string (128)	The description of the package.
height	Yes	integer <int32>	The height of the item in centimeters.
identifier	Yes	string (100)	A unique identifier for the package. This value will be considered when producing verification barcodes.
items	NA	object	The items to be delivered.
length	Yes	integer <int32>	The length of the item in centimeters.
packDimensionCode	NA	string	The package dimension code.
totalAmount	NA	number	The combined prices of all items within the package.
weight	Yes	number	The weight of the item in kilograms.
width	Yes	integer <int32>	The width of the item in centimeters.

The following table describes the elements of an individual item within a delivery.

Table 4-32 Item — Object

Element Name	Required	Data Type	Description
description	Yes	string (128)	The description of the item.
identifier	Yes	string (100)	The identifier (UPC, EAN, SKU, and so on) of the item.
price	NA	number	The price of the item.
quantity	Yes	integer <int32>	The quantity of the item.

Responses

This section describes the responses of the Build Delivery endpoint.

Response Code: 200

The result object.

The media type is application/json.

Example Value

```
{
  "delivery": {
    "created": "2024-11-28T12:33:59.747Z",
    "createdBy": "string",
    "updated": "2024-11-28T12:33:59.747Z",
    "updatedBy": "string",
    "id": <id>,
    "barcodes": [
      {
        "created": "2024-11-28T12:33:59.747Z",
        "createdBy": "string",
        "updated": "2024-11-28T12:33:59.747Z",
        "updatedBy": "string",
        "id": <id>,
        "identifier": "string",
        "parentId": ,
        "type": "CODE39",
        "value": "string"
      }
    ],
    "cancelReasonCode": "string",
    "canceledBy": "string",
    "crossReferences": [
      {
        "created": "2024-11-28T12:33:59.747Z",
        "createdBy": "string",
        "updated": "2024-11-28T12:33:59.747Z",
        "updatedBy": "string",
        "id": <id>,
        "key": "string",
```

```
        "parentId": <parentId>,\n        "value": "string"\n    }\n],\n"currency": "str",\n"deliveryCancelReason": "COURIER",\n"deliveryId": "string",\n"deliveryType": "DELIVERY",\n"dropoffAddress": {\n    "created": "2024-11-28T12:33:59.747Z",\n    "createdBy": "string",\n    "updated": "2024-11-28T12:33:59.747Z",\n    "updatedBy": "string",\n    "id": 9223372036854776000,\n    "address1": "string",\n    "address2": "string",\n    "address3": "string",\n    "address4": "string",\n    "addressType": "RESIDENTIAL",\n    "apartment": "string",\n    "city": "string",\n    "country": "string",\n    "latitude": 999999999,\n    "locationCode": "string",\n    "longitude": 999999999,\n    "postalCode": "string",\n    "state": "string"\n},\n"dropoffAddressId": 9223372036854776000,\n"dropoffContact": {\n    "created": "2024-11-28T12:33:59.747Z",\n    "createdBy": "string",\n    "updated": "2024-11-28T12:33:59.747Z",\n    "updatedBy": "string",\n    "id": <id>,\n    "businessName": "string",\n    "contactReference": "string",\n    "emailAddress": "string",\n    "firstName": "string",\n    "lastName": "string",\n    "middleName": "string",\n    "phoneNumber": "string",\n    "suffix": "string"\n},\n"dropoffContactId": <dropoffContactId>,\n"dropoffDeadline": "2024-11-28T12:33:59.747Z",\n"dropoffEta": "2024-11-28T12:33:59.747Z",\n"dropoffLocationCode": "string",\n"dropoffNotes": "string",\n"dropoffReady": "2024-11-28T12:33:59.747Z",\n"dropoffVerification": {\n    "created": "2024-11-28T12:33:59.747Z",\n    "createdBy": "string",\n    "updated": "2024-11-28T12:33:59.747Z",\n    "updatedBy": "string",\n    "id": <id>,\n
```

```
"barcodeRequired": false,
"barcodes": {
  "additionalProp1": "CODE39",
  "additionalProp2": "CODE39",
  "additionalProp3": "CODE39"
},
"contactRequired": false,
"courierNotes": "string",
"identificationVerified": true,
"minimumAge": 9999,
"pictureRequired": false,
"pictureURL": "string",
"signatureRequired": false,
"signatureURL": "string"
},
"dropoffVerificationId": <dropoffVerificationId>,
"errorCount": 9999,
"fee": 10000000000000000,
"labelsAttached": false,
"manifest": [
  {
    "created": "2024-11-28T12:33:59.747Z",
    "createdBy": "string",
    "updated": "2024-11-28T12:33:59.747Z",
    "updatedBy": "string",
    "id": <id>,
    "description": "string",
    "height": 9999,
    "identifier": "string",
    "items": [
      {
        "created": "2024-11-28T12:33:59.747Z",
        "createdBy": "string",
        "updated": "2024-11-28T12:33:59.747Z",
        "updatedBy": "string",
        "id": <id>,
        "description": "string",
        "identifier": "string",
        "parentId": <parentId>,
        "price": 10000000000000000,
        "quantity": 9999
      }
    ],
    "length": 9999,
    "packDimensionCode": "string",
    "parentId": <parentId>,
    "totalAmount": 10000000000000000,
    "weight": 10000000000000000,
    "width": 9999
  }
],
"manifestSize": 9999,
"pickupAddress": {
  "created": "2024-11-28T12:33:59.747Z",
  "createdBy": "string",
  "updated": "2024-11-28T12:33:59.747Z",
```

```
"updatedBy": "string",
"id": <id>,
"address1": "string",
"address2": "string",
"address3": "string",
"address4": "string",
"addressType": "RESIDENTIAL",
"apartment": "string",
"city": "string",
"country": "string",
"latitude": 999999999,
"locationCode": "string",
"longitude": 999999999,
"postalCode": "string",
"state": "string"
},
"pickupAddressId": <pickupAddressId>,
"pickupContact": {
  "created": "2024-11-28T12:33:59.747Z",
  "createdBy": "string",
  "updated": "2024-11-28T12:33:59.747Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"pickupContactId": <pickupContactId>,
"pickupDeadline": "2024-11-28T12:33:59.747Z",
"pickupEta": "2024-11-28T12:33:59.747Z",
"pickupLocationCode": "string",
"pickupNotes": "string",
"pickupReady": "2024-11-28T12:33:59.747Z",
"pickupVerification": {
  "created": "2024-11-28T12:33:59.747Z",
  "createdBy": "string",
  "updated": "2024-11-28T12:33:59.747Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodeRequired": false,
  "barcodes": {
    "additionalProp1": "CODE39",
    "additionalProp2": "CODE39",
    "additionalProp3": "CODE39"
  },
  "contactRequired": false,
  "courierNotes": "string",
  "identificationVerified": true,
  "minimumAge": 9999,
  "pictureRequired": false,
  "pictureURL": "string",
```

```
        "signatureRequired": false,
        "signatureURL": "string"
    },
    "pickupVerificationId": 9223372036854776000,
    "provider": "string",
    "providerCreated": "2024-11-28T12:33:59.747Z",
    "providerId": "string",
    "providerInstance": "string",
    "returnTripId": 9223372036854776000,
    "status": "CREATED",
    "statusHistory": [
        {
            "created": "2024-11-28T12:33:59.747Z",
            "createdBy": "string",
            "updated": "2024-11-28T12:33:59.747Z",
            "updatedBy": "string",
            "id": <id>,
            "newStatus": "CREATED",
            "parentId": 9223372036854776000,
            "priorStatus": "CREATED"
        }
    ],
    "tip": 1000000000000000000,
    "totalValue": 1000000000000000000,
    "trackingURL": "string",
    "undeliverableInfo": {
        "created": "2024-11-28T12:33:59.747Z",
        "createdBy": "string",
        "updated": "2024-11-28T12:33:59.747Z",
        "updatedBy": "string",
        "id": <id>,
        "moreInfo": {
            "additionalProp1": "string",
            "additionalProp2": "string",
            "additionalProp3": "string"
        },
        "reason": "string",
        "undeliverableAction": "LEAVE_AT_DOOR"
    },
    "undeliverableInfoId": 9223372036854776000
},
"errors": [
    {
        "errorCode": 2147483647,
        "errorMessage": "string",
        "moreInfo": {
            "code": "string",
            "description": "string"
        },
        "provider": "string"
    }
]
```

Schema — Delivery Response

Schema — Delivery Response

The following table describes the elements of a response to a delivery request.

Table 4-33 Delivery Response — Object

Element Name	Required	Data Type	Description
delivery	NA	object	A delivery from a delivery provider.
errors	NA	object	A collection of errors from providers which could not service the request.

The following table describes the elements of a delivery from a delivery provider.

Table 4-34 Delivery — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was last created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
barcodes	NA	object	A list of barcodes associated with the delivery.
cancelReasonCode	NA	string(40)	The user cancel reason code.
canceledBy	NA	string (256)	The ID of the caller who requested cancelation of the delivery (if applicable).
crossReferences	NA	object	A list of cross-references for the delivery.
currency	NA	string (3)	The currency code associated with the delivery.
deliveryCancelReason	NA	string	The delivery cancel reason. Enum: COURIER, PROVIDER, CUSTOMER
deliveryId	Yes	string (128)	The primary reference ID from the calling system uniquely identifying this delivery.
deliveryType	Yes	string	The delivery type identifier. Default: DELIVERY; Enum: DELIVERY, RETURN
dropoffAddress	NA	object	An address for a given dropoff location.

Table 4-34 (Cont.) Delivery — Object

Element Name	Required	Data Type	Description
dropoffAddressId	NA	integer <int64>	NA
dropoffContact	NA	object	A point of contact for a given dropoff location.
dropoffContactId	NA	integer <int64>	NA
dropoffDeadline	NA	string <date-time>	The time by which the order must be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.
dropoffEta	NA	string <date-time>	The time by which the order is expected to be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.
dropoffLocationCode	NA	string(10)	The location code for the dropoff address.
dropoffNotes	NA	string(256)	Notes for the courier at dropoff location.
dropoffReady	NA	string(\$date-time)	The time at which the items may be dropped off.
dropoffVerification	NA	object	A record indicating verification proof information for a given delivery.
dropoffVerificationId	NA	integer <int64>	NA
errorCount	NA	integer(\$int32)	The count of failed Provider API calls.
fee	NA	number	The cost of the delivery.
labelsAttached	NA	boolean	Indicates whether the package(s) will have a pre-paid shipping label attached when the courier arrives. Default: false
manifest		object	The packages to be delivered.
manifestSize	NA	integer <int32>	The count of manifest items in the delivery.
pickupAddress	NA	object	An address for a given pickup location.
pickupAddressId	NA	integer <int64>	NA
pickupContact	NA	object	A point of contact for a given pickup location.
pickupContactId	NA	integer <int64>	NA
pickupDeadline	NA	string <date-time>	The time by which the order must be picked up. Date-time strings are in ISO 8601 timestamp with offset format.

Table 4-34 (Cont.) Delivery — Object

Element Name	Required	Data Type	Description
pickupEta	NA	string <date-time>	The time by which the order is expected to be picked up. Date-time strings are in ISO 8601 timestamp with offset format.
pickupLocationCode	NA	string (10)	The pickup location code associated with this delivery.
pickupNotes	NA	string(256)	Notes for the courier at pickup location.
pickupReady	NA	string(\$date-time)	The time at which the items will be ready for pickup.
pickupVerification	NA	object	A record indicating verification proof information for a given delivery.
pickupVerificationId	NA	integer <int64>	NA
provider	Yes	string (10)	The provider of the delivery.
providerCreated	Yes	string <date-time>	The creation timestamp, as indicated by the provider. Date-time strings are in ISO 8601 timestamp with offset format.
providerInstance	Yes	string (20)	The specific instance of the provider of the delivery.
providerId	Yes	string (128)	The id from the provider of the delivery.
returnTripId	NA	integer(\$int64)	The return trip id of the delivery.
status	Yes	string	The last known status of the delivery. enum: CREATED, PENDING, ENROUTE_PICKUP, PICKUP_COMPLETE, ENROUTE_DROPOFF, DROPOFF_COMPLETE, CANCELED, RETURNED, ERROR, ENROUTE_RETURN
statusHistory	NA	object	Status history for the delivery.
tip	NA	number	The amount of the tip.
totalValue	NA	number	The total value of the request.
trackingURL	NA	string	The tracking URL for the delivery.
undeliverableInfo	NA	object	A record indicating details about a delivery that could not be completed.
undeliverableInfoId	NA	integer <int64>	NA

The following table describes elements of a barcode for a delivery package.

Table 4-35 Manifest Package Barcode — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	NA
identifier	Yes	string (100)	The identifier for the manifest entry associated with this barcode.
parentId	NA	integer <int64>	NA
type	NA	string	enum: CODE39, CODE39_FULL_ASCII, CODE128, QR
value	Yes	string (100)	The the value of the barcode.

The following table describes the elements of a cross reference record.

Table 4-36 Delivery Cross References — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
key	Yes	string (128)	The identifier for the cross reference.
parentId	NA	integer <int64>	Parent ID
value	Yes	string (128)	The the value of the cross reference.

The following table describes the elements of an address for a given pickup or delivery.

Table 4-37 Address

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.

Table 4-37 (Cont.) Address

Element Name	Required	Data Type	Description
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
address1	Yes	string (254)	The address line 1.
address2	NA	string (254)	The address line 2.
address3	NA	string (254)	The address line 3.
address4	NA	string (254)	The address line 4.
addressType	NA	string	enum: RESIDENTIAL, BUSINESS
apartment	NA	string (254)	The apartment.
city	Yes	string (254)	The city or town.
country	Yes	string (254)	The ISO 3166 country code. Applications can decide whether to use 2 or 3 char codes.
latitude	NA	number	The latitude of the address.
locationCode	NA	string (10)	The location code for the address, if applicable. This must be unique across the retailer's landscape.
longitude	NA	number	The longitude of the address.
postalCode	NA	string (20)	The postal code.
state	NA	string (254)	The ISO 3166 state, province or territory code.

The following table describes the elements of a point of contact for a given pickup or delivery.

Table 4-38 Contact

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <\$date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
businessName	NA	string (254)	The business name.
contactReference	NA	string (64)	A reference number for the contact.

Table 4-38 (Cont.) Contact

Element Name	Required	Data Type	Description
emailAddress	NA	string (254)	The email address.
firstName	NA	string (254)	The first (that is, given) name.
lastName	NA	string (254)	The last (that is, family) name.
middleName	NA	string (254)	The middle name.
phoneNumber	Yes	string (50)	The phone number.
suffix	NA	string (254)	A name suffix.

The following table describes a record indicating details for a delivery package.

Table 4-39 Manifest Package Detail — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
description	NA	string(256)	The description of the package.
height	NA	integer(\$int32)	The height of the item in centimeters.
identifier	Yes	string(100)	The identifier for the manifest package entry.
items	NA	object	The packages to be delivered.
length	NA	integer(\$int32)	The length of the item in centimeters.
packDimensionCode	NA	string(15)	The package dimension code.
parentId	NA	integer(\$int64)	The parent ID.
totalAmount	NA	number	The combined prices of all items within the package.
weight	NA	number	The weight of the item in kilograms.
width	NA	integer(\$int32)	The width of the item in centimeters.

The following table describes a record indicating details for an item in the delivery package.

Table 4-40 Manifest Item Detail — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.

Table 4-40 (Cont.) Manifest Item Detail — Object

Element Name	Required	Data Type	Description
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
description	NA	string(256)	The description of the item.
height	NA	integer(\$int32)	The height of the item in centimeters.
identifier	Yes	string(100)	The identifier for the item entry.
parentId	NA	integer(\$int64)	The parent ID.
price	NA	number	The price of the item.
quantity	NA	integer(\$int32)	The quantity of the item.

The following table describes elements of a record indicating verification proof information for a given delivery.

Table 4-41 Delivery Verification Proof — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
barcodeRequired	NA	boolean	Indicates that a barcode scan is required. Default: false
barcodes	NA	string	Barcode values and types that were scanned. Enum: CODE39, CODE39_FULL_ASCII, CODE128, QR
contactRequired	NA	boolean	Indicates that contact is required (that is, must NOT be contactless). Default: false
courierNotes	NA	string (2000)	Additional notes pertaining to proof of handoff.
identificationVerified	NA	boolean	Indicates that identification was successfully verified.
minimumAge	NA	integer(\$int32)	The minimum age of the recipient.

Table 4-41 (Cont.) Delivery Verification Proof — Object

Element Name	Required	Data Type	Description
pictureRequired	NA	boolean	Indicates that a picture is required. Default: false
pictureURL	NA	string	The URL for the picture image.
signatureRequired	NA	boolean	Indicates that a signature is required. Default: false
signatureURL	NA	string	The URL for the signature image.

The following table describes elements of a record indicating a status transition for a delivery.

Table 4-42 Delivery Status History — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
newStatus	Yes	string	The new status of the delivery. Enum: CREATED, PENDING, ENROUTE_PICKUP, PICKUP_COMPLETE, ENROUTE_DROPOFF, DROPOFF_COMPLETE, CANCELED, RETURNED, ERROR, ENROUTE_RETURN
parentId	NA	integer <int64>	Parent ID
priorStatus	Yes	string	The prior status of the delivery. Enum: CREATED, PENDING, ENROUTE_PICKUP, PICKUP_COMPLETE, ENROUTE_DROPOFF, DROPOFF_COMPLETE, CANCELED, RETURNED, ERROR, ENROUTE_RETURN

The following table describes elements of a record indicating details about a delivery that could not be completed.

Table 4-43 Delivery Undeliverable Information — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
moreInfo	NA	string	Any additional provider-specific information which might be available.
reason	NA	string (2000)	The reason the delivery was undeliverable.
undeliverableAction	NA	string	Indicates what the courier should do with the packages in the event they are unable to be delivered. Enum: DISCARD, LEAVE_AT_DOOR, RETURN

The following table describes elements of a container for an error from a provider.

Table 4-44 Provider Error — Object

Element Name	Required	Data Type	Description
errorCode	NA	integer <int32>	A numeric error indicating the type of error.
errorMessage	NA	string	An unlocalized message indicating details about the error.
moreInfo	NA	object	An object containing provider-specific details pertaining to an error.
provider	Yes	string	The provider associated with the error.

The following table describes elements of an object containing provider-specific details pertaining to an error.

Table 4-45 More Error Information — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Response Code: 204

No providers were eligible to service this request.

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

For more information about the Mapper Error Data schema, see the [Schema — Mapper Error Data](#) section.

Response Code: 404

Provided quote ID or associated provider could not be found.

Response Code: 409

Provided delivery ID already exists.

Response Code: 424

No providers could service this request.

The media type is application/json.

Example Value

```
[
  {
    "errorCode": 2147483647,
    "errorMessage": "string",
  }
]
```

```
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "provider": "string"
  }
]
```

Schema — Provider Error

For more information about the Provider Error schema, see the [Schema — Provider Error](#) section.

Response Code: 500

The request failed.

Cancel a Delivery — POST

This section describes the Cancel a Delivery endpoint. This service attempts to cancel the delivery, contacting the provider to do so.

Method

POST

URL

/api/delivery/v5/delivery/cancel

Allowed Scope(s)

delivery-rw

Request

This section describes the request parameters.

Table 4-46 Cancel a Delivery — Request Parameters

Parameter	Required	Data Type	Description
id	NA	integer <int64> (query)	The identifier of the delivery.
deliveryId	NA	string (query)	The primary reference ID from the originating system uniquely identifying this delivery.

Table 4-46 (Cont.) Cancel a Delivery — Request Parameters

Parameter	Required	Data Type	Description
cancelReasonCode	NA	string (query)	The user cancel reason code.

Responses

This section describes the responses of the Cancel a Delivery endpoint.

Response Code: 200

The result object.

The media type is application/json.

Example Value

```
{
  "created": "2024-11-28T13:11:28.299Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:11:28.299Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodes": [
    {
      "created": "2024-11-28T13:11:28.299Z",
      "createdBy": "string",
      "updated": "2024-11-28T13:11:28.299Z",
      "updatedBy": "string",
      "id": <id>,
      "identifier": "string",
      "parentId": <parentId>,
      "type": "CODE39",
      "value": "string"
    }
  ],
  "cancelReasonCode": "string",
  "canceledBy": "string",
  "crossReferences": [
    {
      "created": "2024-11-28T13:11:28.299Z",
      "createdBy": "string",
      "updated": "2024-11-28T13:11:28.299Z",
      "updatedBy": "string",
      "id": <id>,
      "key": "string",
      "parentId": <parentId>,
      "value": "string"
    }
  ],
  "currency": "str",
```

```
"deliveryCancelReason": "COURIER",
"deliveryId": "string",
"deliveryType": "DELIVERY",
"dropoffAddress": {
  "created": "2024-11-28T13:11:28.299Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:11:28.299Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"dropoffAddressId": <dropoffAddressId>,
"dropoffContact": {
  "created": "2024-11-28T13:11:28.299Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:11:28.299Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"dropoffContactId": <dropoffContactId>,
"dropoffDeadline": "2024-11-28T13:11:28.299Z",
"dropoffEta": "2024-11-28T13:11:28.299Z",
"dropoffLocationCode": "string",
"dropoffNotes": "string",
"dropoffReady": "2024-11-28T13:11:28.299Z",
"dropoffVerification": {
  "created": "2024-11-28T13:11:28.299Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:11:28.299Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodeRequired": false,
  "barcodes": {
    "additionalProp1": "CODE39",
    "additionalProp2": "CODE39",
    "additionalProp3": "CODE39"
```

```
    },
    "contactRequired": false,
    "courierNotes": "string",
    "identificationVerified": true,
    "minimumAge": 9999,
    "pictureRequired": false,
    "pictureURL": "string",
    "signatureRequired": false,
    "signatureURL": "string"
  },
  "dropoffVerificationId": <dropoffVerificationId>,
  "errorCount": 9999,
  "fee": 10000000000000000,
  "labelsAttached": false,
  "manifest": [
    {
      "created": "2024-11-28T13:11:28.299Z",
      "createdBy": "string",
      "updated": "2024-11-28T13:11:28.299Z",
      "updatedBy": "string",
      "id": <id>,
      "description": "string",
      "height": 9999,
      "identifier": "string",
      "items": [
        {
          "created": "2024-11-28T13:11:28.299Z",
          "createdBy": "string",
          "updated": "2024-11-28T13:11:28.299Z",
          "updatedBy": "string",
          "id": <id>,
          "description": "string",
          "identifier": "string",
          "parentId": <parentId>,
          "price": 10000000000000000,
          "quantity": 9999
        }
      ],
      "length": 9999,
      "packDimensionCode": "string",
      "parentId": <parentId>,
      "totalAmount": 10000000000000000,
      "weight": 10000000000000000,
      "width": 9999
    }
  ],
  "manifestSize": 9999,
  "pickupAddress": {
    "created": "2024-11-28T13:11:28.299Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:11:28.299Z",
    "updatedBy": "string",
    "id": <id>,
    "address1": "string",
    "address2": "string",
    "address3": "string",
```

```
"address4": "string",
"addressType": "RESIDENTIAL",
"apartment": "string",
"city": "string",
"country": "string",
"latitude": 999999999,
"locationCode": "string",
"longitude": 999999999,
"postalCode": "string",
"state": "string"
},
"pickupAddressId": <pickAddressId>,
"pickupContact": {
  "created": "2024-11-28T13:11:28.299Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:11:28.299Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"pickupContactId": <pickupContactId>,
"pickupDeadline": "2024-11-28T13:11:28.299Z",
"pickupEta": "2024-11-28T13:11:28.299Z",
"pickupLocationCode": "string",
"pickupNotes": "string",
"pickupReady": "2024-11-28T13:11:28.299Z",
"pickupVerification": {
  "created": "2024-11-28T13:11:28.299Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:11:28.299Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodeRequired": false,
  "barcodes": {
    "additionalProp1": "CODE39",
    "additionalProp2": "CODE39",
    "additionalProp3": "CODE39"
  },
  "contactRequired": false,
  "courierNotes": "string",
  "identificationVerified": true,
  "minimumAge": 9999,
  "pictureRequired": false,
  "pictureURL": "string",
  "signatureRequired": false,
  "signatureURL": "string"
},
"pickupVerificationId": <pickupVerificationId>,
"provider": "string",
```

```
"providerCreated": "2024-11-28T13:11:28.299Z",
"providerId": "string",
"providerInstance": "string",
"returnTripId": 9223372036854776000,
"status": "CREATED",
"statusHistory": [
  {
    "created": "2024-11-28T13:11:28.299Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:11:28.299Z",
    "updatedBy": "string",
    "id": <id>,
    "newStatus": "CREATED",
    "parentId": <parentId>,
    "priorStatus": "CREATED"
  }
],
"tip": 1000000000000000000,
"totalValue": 1000000000000000000,
"trackingURL": "string",
"undeliverableInfo": {
  "created": "2024-11-28T13:11:28.299Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:11:28.299Z",
  "updatedBy": "string",
  "id": <id>,
  "moreInfo": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "reason": "string",
  "undeliverableAction": "LEAVE_AT_DOOR"
},
"undeliverableInfoId": <undeliverableInfoId>
}
```

Schema — Delivery

Schema — Delivery

The following table describes the elements of a delivery from a delivery provider.

Table 4-47 Delivery — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.

Table 4-47 (Cont.) Delivery — Object

Element Name	Required	Data Type	Description
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
barcodes	NA	object	A list of barcodes associated with the delivery.
cancelReasonCode	NA	string(40)	The user cancel reason code.
canceledBy	NA	string (256)	The ID of the caller who requested cancellation of the delivery (if applicable).
crossReferences	NA	object	A list of cross-references for the delivery.
currency	NA	string (3)	The currency code associated with the delivery.
deliveryCancelReason	NA	string	The delivery cancel reason. Enum: COURIER, PROVIDER, CUSTOMER
deliveryId	Yes	string (128)	The primary reference ID from the calling system uniquely identifying this delivery.
deliveryType	Yes	string	The delivery identifier. Default: DELIVERY; Enum: DELIVERY, RETURN
dropoffAddress	NA	object	An address for a given dropoff location.
dropoffAddressId	NA	integer <int64>	NA
dropoffContact	NA	object	A point of contact for a given dropoff location.
dropoffContactId	NA	integer <int64>	NA
dropoffDeadline	NA	string <date-time>	The time by which the order must be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.
dropoffEta	NA	string <date-time>	The time by which the order is expected to be dropped off. Date-time strings are in ISO 8601 timestamp with offset format.
dropoffLocationCode	NA	string(10)	The location code for the dropoff address.
dropoffNotes	NA	string(256)	Notes for the courier at dropoff location.
dropoffReady	NA	string(\$date-time)	The time at which the items may be dropped off.
dropoffVerification	NA	object	A record indicating verification proof information for a given delivery.
dropoffVerificationId	NA	integer <int64>	NA
errorCount	NA	integer(\$int32)	The count of failed Provider API calls.

Table 4-47 (Cont.) Delivery — Object

Element Name	Required	Data Type	Description
fee	NA	number	The cost of the delivery.
labelsAttached	NA	boolean	Indicates whether the package(s) will have a pre-paid shipping label attached when the courier arrives. Default: false Note: Return delivery types have the following limitations with the Uber integration: Labels Attached will always be set to TRUE which indicates that a label will be attached to the package(s).
manifest	NA	object	The packages to be delivered.
manifestSize	NA	integer <int32>	The count of manifest items in the delivery.
pickupAddress	NA	object	An address for a given pickup location.
pickupAddressId	NA	integer <int64>	NA
pickupContact	NA	object	A point of contact for a given pickup location.
pickupContactId	NA	integer <int64>	NA
pickupDeadline	NA	string <date-time>	The time by which the order must be picked up. Date-time strings are in ISO 8601 timestamp with offset format.
pickupEta	NA	string <date-time>	The time by which the order is expected to be picked up. Date-time strings are in ISO 8601 timestamp with offset format.
pickupLocationCode	NA	string (10)	The pickup location code associated with this delivery.
pickupNotes	NA	string(256)	Notes for the courier at pickup location.
pickupReady	NA	string(\$date-time)	The time at which the items will be ready for pickup.
pickupVerification	NA	object	A record indicating verification proof information for a given delivery.
pickupVerificationId	NA	integer <int64>	NA
provider	Yes	string (10)	The provider of the delivery.
providerCreated	Yes	string <date-time>	The creation timestamp, as indicated by the provider. Date-time strings are in ISO 8601 timestamp with offset format.
providerInstance	Yes	string (20)	The specific instance of the provider of the delivery.
providerId	Yes	string (128)	The id from the provider of the delivery.
returnTripId	NA	integer(\$int64)	The return trip id of the delivery.

Table 4-47 (Cont.) Delivery — Object

Element Name	Required	Data Type	Description
status	Yes	string	The last known status of the delivery. enum: CREATED, PENDING, ENROUTE_PICKUP, PICKUP_COMPLETE, ENROUTE_DROPOFF, DROPOFF_COMPLETE, CANCELED, RETURNED, ERROR, ENROUTE_RETURN
statusHistory	NA	object	Status history for the delivery.
tip	NA	number	The amount of the tip.
totalValue	NA	number	The total value of the request.
trackingURL	NA	string	The tracking URL for the delivery.
undeliverableInfo	NA	object	A record indicating details about a delivery that could not be completed.
undeliverableInfo Id	NA	integer <int64>	NA

The following table describes elements of a barcode for a delivery package.

Table 4-48 Manifest Package Barcode — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	NA
identifier	Yes	string (100)	The identifier for the manifest entry associated with this barcode.
parentId	NA	integer <int64>	NA
type	NA	string	enum: CODE39, CODE39_FULL_ASCII, CODE128, QR
value	Yes	string (100)	The the value of the barcode.

The following table describes the elements of a cross reference record.

Table 4-49 Delivery Cross References — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
key	Yes	string (128)	The identifier for the cross reference.
parentId	NA	integer <int64>	Parent ID
value	Yes	string (128)	The the value of the cross reference.

The following table describes the elements of an address for a given pickup or delivery.

Table 4-50 Address

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
address1	Yes	string (254)	The address line 1.
address2	NA	string (254)	The address line 2.
address3	NA	string (254)	The address line 3.
address4	NA	string (254)	The address line 4.
addressType	NA	string	enum: RESIDENTIAL, BUSINESS
apartment	NA	string (254)	The apartment.
city	Yes	string (254)	The city or town.
country	Yes	string (254)	The ISO 3166 country code. Applications can decide whether to use 2 or 3 char codes.
latitude	NA	number	The latitude of the address.
locationCode	NA	string (10)	The location code for the address, if applicable. This must be unique across the retailer's landscape.

Table 4-50 (Cont.) Address

Element Name	Required	Data Type	Description
longitude	NA	number	The longitude of the address.
postalCode	NA	string (20)	The postal code.
state	NA	string (254)	The ISO 3166 state, province or territory code.

The following table describes the elements of a point of contact for a given pickup or delivery.

Table 4-51 Contact

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
businessName	NA	string (254)	The business name.
contactReference	NA	string (64)	A reference number for the contact.
emailAddress	NA	string (254)	The email address.
firstName	NA	string (254)	The first (that is, given) name.
lastName	NA	string (254)	The last (that is, family) name.
middleName	NA	string (254)	The middle name.
phoneNumber	Yes	string (50)	The phone number.
suffix	NA	string (254)	A name suffix.

The following table describes elements of a record indicating verification proof information for a given delivery.

Table 4-52 Delivery Verification Proof — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.

Table 4-52 (Cont.) Delivery Verification Proof — Object

Element Name	Required	Data Type	Description
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
barcodeRequired	NA	boolean	Indicates that a barcode scan is required. Default: false
barcodes	NA	string	Barcode values and types that were scanned. Enum: CODE39, CODE39_FULL_ASCII, CODE128, QR
contactRequired	NA	boolean	Indicates that contact is required (that is, must NOT be contactless) Default: false
courierNotes	NA	string (2000)	Additional notes pertaining to proof of handoff.
identificationVerified	NA	boolean	Indicates that identification was successfully verified.
minimumAge	NA	integer(\$int32)	The minimum age of the recipient.
pictureRequired	NA	boolean	Indicates that a picture is required. Default: false
pictureURL	NA	string	The URL for the picture image.
signatureRequired	NA	boolean	Indicates that a signature is required. Default: false
signatureURL	NA	string	The URL for the signature image.

The following table describes elements of a record indicating a status transition for a delivery.

Table 4-53 Manifest Package Detail — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
description	NA	string	The description of the package.
height	NA	integer(\$int32)	The height of the item in centimeters.
identifier	Yes	string	The identifier for the manifest package entry.
items	NA	object	The packages to be delivered.
length	NA	integer(\$int32)	The length of the item in centimeters.
packDimensionCode	NA	string(15)	The package dimension code.

Table 4-53 (Cont.) Manifest Package Detail — Object

Element Name	Required	Data Type	Description
parentId	NA	integer(\$int64)	The parent ID.
totalAmount	NA	number	The combined prices of all items within the package.
weight	NA	number	The weight of the item in kilograms.
width	NA	integer(\$int32)	The width of the item in centimeters.

The following table describes a record indicating details for an item in the delivery package.

Table 4-54 Manifest Item Detail — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
description	NA	string(256)	The description of the item.
height	NA	integer(\$int32)	The height of the item in centimeters.
identifier	Yes	string(100)	The identifier for the item entry.
parentId	NA	integer(\$int64)	The parent ID.
price	NA	number	The price of the item.
quantity	NA	integer(\$int32)	The quantity of the item.

Table 4-55 Delivery Status History — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID

Table 4-55 (Cont.) Delivery Status History — Object

Element Name	Required	Data Type	Description
newStatus	Yes	string	The new status of the delivery. Enum: CREATED, PENDING, ENROUTE_PICKUP, PICKUP_COMPLETE, ENROUTE_DROPOFF, DROPOFF_COMPLETE, CANCELED, RETURNED, ERROR, ENROUTE_RETURN
parentId	NA	integer <int64>	Parent ID
priorStatus	Yes	string	The prior status of the delivery. Enum: CREATED, PENDING, ENROUTE_PICKUP, PICKUP_COMPLETE, ENROUTE_DROPOFF, DROPOFF_COMPLETE, CANCELED, RETURNED, ERROR, ENROUTE_RETURN

The following table describes elements of a record indicating details about a delivery that could not be completed.

Table 4-56 Delivery Undeliverable Information — Object

Element Name	Required	Data Type	Description
created	NA	string <date-time>	The timestamp at which the record was created. Date-time strings are in ISO 8601 timestamp with offset format.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string <date-time>	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer <int64>	ID
moreInfo	NA	string	Any additional provider-specific information which might be available.
reason	NA	string (2000)	The reason the delivery was undeliverable.
undeliverableAction	NA	string	Indicates what the courier should do with the packages in the event they are unable to be delivered. Enum: DISCARD, LEAVE_AT_DOOR, RETURN

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

For more information about the schema Mapper Error Data, see the [Schema — Mapper Error Data](#) section.

Response Code: 404

Provided delivery ID could not be found.

Response Code: 409

Provided delivery ID is not in a status eligible for cancellation.

Response Code: 424

No provider could service this request.

The media type is application/json.

Example Value

```
[
  {
    "errorCode": 2147483647,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "provider": "string"
  }
]
```



```
}  
]
```

Schema — Provider Error

For more information about the Provider Error Schema, see the [Schema — Provider Error](#) section.

Response Code: 500

The request failed.

Get the Current Status of a Delivery — GET

This section describes the Get the Current Status of a Delivery endpoint. This service retrieves the latest status of the delivery, calling the provider if appropriate. It will also retrieve the most current details about delivery in case of updates to the delivery schedule, notes, or other.

Method

GET

URL

/api/delivery/v5/delivery/status

Allowed Scope(s)

delivery-r
delivery-rw

Request

This section describes the request parameters.

Table 4-57 Get the Current Status of a Delivery — Request Parameters

Parameter	Required	Data Type	Description
id	NA	integer <int64> (query)	The identifier of the delivery.
deliveryId	NA	string (query)	The primary reference ID from the originating system uniquely identifying this delivery.

Responses

This section describes responses of the Get the Current Status of a Delivery endpoint.

Response Code: 200

The result object.

The media type is application/json.

Example Value

```
{
  "created": "2024-11-28T13:28:27.877Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:28:27.877Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodes": [
    {
      "created": "2024-11-28T13:28:27.877Z",
      "createdBy": "string",
      "updated": "2024-11-28T13:28:27.877Z",
      "updatedBy": "string",
      "id": <id>,
      "identifier": "string",
      "parentId": <parentId>,
      "type": "CODE39",
      "value": "string"
    }
  ],
  "cancelReasonCode": "string",
  "canceledBy": "string",
  "crossReferences": [
    {
      "created": "2024-11-28T13:28:27.877Z",
      "createdBy": "string",
      "updated": "2024-11-28T13:28:27.877Z",
      "updatedBy": "string",
      "id": <id>,
      "key": "string",
      "parentId": <parentId>,
      "value": "string"
    }
  ],
  "currency": "str",
  "deliveryCancelReason": "COURIER",
  "deliveryId": "string",
  "deliveryType": "DELIVERY",
  "dropoffAddress": {
    "created": "2024-11-28T13:28:27.877Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:28:27.877Z",
    "updatedBy": "string",
    "id": <id>,
    "address1": "string",
    "address2": "string",
    "address3": "string",
```

```
"address4": "string",
"addressType": "RESIDENTIAL",
"apartment": "string",
"city": "string",
"country": "string",
"latitude": 999999999,
"locationCode": "string",
"longitude": 999999999,
"postalCode": "string",
"state": "string"
},
"dropoffAddressId": <dropoffAddressId>,
"dropoffContact": {
  "created": "2024-11-28T13:28:27.877Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:28:27.877Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"dropoffContactId": <dropoffContactId>,
"dropoffDeadline": "2024-11-28T13:28:27.877Z",
"dropoffEta": "2024-11-28T13:28:27.877Z",
"dropoffLocationCode": "string",
"dropoffNotes": "string",
"dropoffReady": "2024-11-28T13:28:27.877Z",
"dropoffVerification": {
  "created": "2024-11-28T13:28:27.877Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:28:27.877Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodeRequired": false,
  "barcodes": {
    "additionalProp1": "CODE39",
    "additionalProp2": "CODE39",
    "additionalProp3": "CODE39"
  },
  "contactRequired": false,
  "courierNotes": "string",
  "identificationVerified": true,
  "minimumAge": 9999,
  "pictureRequired": false,
  "pictureURL": "string",
  "signatureRequired": false,
  "signatureURL": "string"
},
"dropoffVerificationId": <dropoffVerificationId>,
"errorCount": 9999,
```

```
"fee": 10000000000000000,
"labelsAttached": false,
"manifest": [
  {
    "created": "2024-11-28T13:28:27.877Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:28:27.877Z",
    "updatedBy": "string",
    "id": <id>,
    "description": "string",
    "height": 9999,
    "identifier": "string",
    "items": [
      {
        "created": "2024-11-28T13:28:27.877Z",
        "createdBy": "string",
        "updated": "2024-11-28T13:28:27.877Z",
        "updatedBy": "string",
        "id": <id>,
        "description": "string",
        "identifier": "string",
        "parentId": 9223372036854776000,
        "price": 10000000000000000,
        "quantity": 9999
      }
    ],
    "length": 9999,
    "packDimensionCode": "string",
    "parentId": <parentId>,
    "totalAmount": 10000000000000000,
    "weight": 10000000000000000,
    "width": 9999
  }
],
"manifestSize": 9999,
"pickupAddress": {
  "created": "2024-11-28T13:28:27.877Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:28:27.877Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"pickupAddressId": <pickupAddressId>
```

```
"pickupContact": {
  "created": "2024-11-28T13:28:27.877Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:28:27.877Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"pickupContactId": <pickupContactId>,
"pickupDeadline": "2024-11-28T13:28:27.877Z",
"pickupEta": "2024-11-28T13:28:27.877Z",
"pickupLocationCode": "string",
"pickupNotes": "string",
"pickupReady": "2024-11-28T13:28:27.877Z",
"pickupVerification": {
  "created": "2024-11-28T13:28:27.877Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:28:27.877Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodeRequired": false,
  "barcodes": {
    "additionalProp1": "CODE39",
    "additionalProp2": "CODE39",
    "additionalProp3": "CODE39"
  },
  "contactRequired": false,
  "courierNotes": "string",
  "identificationVerified": true,
  "minimumAge": 9999,
  "pictureRequired": false,
  "pictureURL": "string",
  "signatureRequired": false,
  "signatureURL": "string"
},
"pickupVerificationId": <pickupVerificationId>,
"provider": "string",
"providerCreated": "2024-11-28T13:28:27.877Z",
"providerId": "string",
"providerInstance": "string",
"returnTripId": 9223372036854776000,
"status": "CREATED",
"statusHistory": [
  {
    "created": "2024-11-28T13:28:27.877Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:28:27.877Z",
    "updatedBy": "string",
    "id": <id>
```

```

        "newStatus": "CREATED",
        "parentId": <parentId>,
        "priorStatus": "CREATED"
    }
],
"tip": 10000000000000000,
"totalValue": 10000000000000000,
"trackingURL": "string",
"undeliverableInfo": {
    "created": "2024-11-28T13:28:27.877Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:28:27.877Z",
    "updatedBy": "string",
    "id": <id>,
    "moreInfo": {
        "additionalProp1": "string",
        "additionalProp2": "string",
        "additionalProp3": "string"
    },
    "reason": "string",
    "undeliverableAction": "LEAVE_AT_DOOR"
},
"undeliverableInfoId": <undeliverableInfoId>
}

```

Schema — Delivery

For more information about the Delivery schema, see the [Schema — Delivery](#) section.

Response Code: 400

The provided input is invalid

The media type is application/json.

Example Value

```

[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]

```

Schema — Mapper Error Data

For more information about the schema Mapper Error Data, see the [Schema — Mapper Error Data](#) section.

Response Code: 404

Provided delivery ID could not be found.

Response Code: 424

No providers could service this request. A collection of related ProviderError objects will be returned.

Example Value

```
[
  {
    "errorCode": 2147483647,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "provider": "string"
  }
]
```

Schema — Provider Error

For more information about the Provider Error schema, see the [Schema — Provider Error](#) section.

Response Code: 500

The request failed.

Get a Delivery — GET

This section describes the Get a Delivery endpoint. This service retrieves the details of a delivery as stored in the database. It will not call out to the provider to retrieve any new updates.

Method

GET

URL

```
/api/delivery/v5/delivery/{id}
```

Allowed Scope(s)

```
delivery-r  
delivery-rw
```

Request

This section describes the request parameters.

Table 4-58 Get a Delivery — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer <int64> (path)	The identifier of the delivery.
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Responses

This section describes the responses of the Get a Delivery endpoint.

Response Code: 200

The requested record.
The media type is application/json.

Example Value

```
{  
  "created": "2024-11-28T13:39:37.057Z",  
  "createdBy": "string",  
  "updated": "2024-11-28T13:39:37.057Z",  
  "updatedBy": "string",  
  "id": <id>,  
  "barcodes": [  

```



```
{
  "created": "2024-11-28T13:39:37.057Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:39:37.057Z",
  "updatedBy": "string",
  "id": <Id>,
  "identifier": "string",
  "parentId": <parentId>,
  "type": "CODE39",
  "value": "string"
}
],
"cancelReasonCode": "string",
"cancelledBy": "string",
"crossReferences": [
  {
    "created": "2024-11-28T13:39:37.057Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:39:37.057Z",
    "updatedBy": "string",
    "id": <id>,
    "key": "string",
    "parentId": <parentId>,
    "value": "string"
  }
],
"currency": "str",
"deliveryCancelReason": "COURIER",
"deliveryId": "string",
"deliveryType": "DELIVERY",
"dropoffAddress": {
  "created": "2024-11-28T13:39:37.058Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:39:37.058Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"dropoffAddressId": <dropoffAddressId>,
"dropoffContact": {
  "created": "2024-11-28T13:39:37.058Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:39:37.058Z",
  "updatedBy": "string",
```

```
"id": <id>,
"businessName": "string",
"contactReference": "string",
"emailAddress": "string",
"firstName": "string",
"lastName": "string",
"middleName": "string",
"phoneNumber": "string",
"suffix": "string"
},
"dropoffContactId": <dropoffContactId>,
"dropoffDeadline": "2024-11-28T13:39:37.058Z",
"dropoffEta": "2024-11-28T13:39:37.058Z",
"dropoffLocationCode": "string",
"dropoffNotes": "string",
"dropoffReady": "2024-11-28T13:39:37.058Z",
"dropoffVerification": {
  "created": "2024-11-28T13:39:37.058Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:39:37.058Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodeRequired": false,
  "barcodes": {
    "additionalProp1": "CODE39",
    "additionalProp2": "CODE39",
    "additionalProp3": "CODE39"
  },
  "contactRequired": false,
  "courierNotes": "string",
  "identificationVerified": true,
  "minimumAge": 9999,
  "pictureRequired": false,
  "pictureURL": "string",
  "signatureRequired": false,
  "signatureURL": "string"
},
"dropoffVerificationId": <dropoffVerificationId>,
"errorCount": 9999,
"fee": 10000000000000000,
"labelsAttached": false,
"manifest": [
  {
    "created": "2024-11-28T13:39:37.058Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:39:37.058Z",
    "updatedBy": "string",
    "id": <id>,
    "description": "string",
    "height": 9999,
    "identifier": "string",
    "items": [
      {
        "created": "2024-11-28T13:39:37.058Z",
        "createdBy": "string",
        "updated": "2024-11-28T13:39:37.058Z",
```

```

        "updatedBy": "string",
        "id": <id>,
        "description": "string",
        "identifier": "string",
        "parentId": <parentId>,
        "price": 10000000000000000,
        "quantity": 9999
    }
},
"length": 9999,
"packDimensionCode": "string",
"parentId": <parentId>,
"totalAmount": 10000000000000000,
"weight": 10000000000000000,
"width": 9999
}
],
"manifestSize": 9999,
"pickupAddress": {
    "created": "2024-11-28T13:39:37.058Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:39:37.058Z",
    "updatedBy": "string",
    "id": <id>,
    "address1": "string",
    "address2": "string",
    "address3": "string",
    "address4": "string",
    "addressType": "RESIDENTIAL",
    "apartment": "string",
    "city": "string",
    "country": "string",
    "latitude": 999999999,
    "locationCode": "string",
    "longitude": 999999999,
    "postalCode": "string",
    "state": "string"
},
"pickupAddressId": <pickupAddressId>,
"pickupContact": {
    "created": "2024-11-28T13:39:37.058Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:39:37.058Z",
    "updatedBy": "string",
    "id": <id>,
    "businessName": "string",
    "contactReference": "string",
    "emailAddress": "string",
    "firstName": "string",
    "lastName": "string",
    "middleName": "string",
    "phoneNumber": "string",
    "suffix": "string"
},
"pickupContactId": <pickupContactId>,
"pickupDeadline": "2024-11-28T13:39:37.058Z",

```

```
"pickupEta": "2024-11-28T13:39:37.058Z",
"pickupLocationCode": "string",
"pickupNotes": "string",
"pickupReady": "2024-11-28T13:39:37.058Z",
"pickupVerification": {
  "created": "2024-11-28T13:39:37.058Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:39:37.058Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodeRequired": false,
  "barcodes": {
    "additionalProp1": "CODE39",
    "additionalProp2": "CODE39",
    "additionalProp3": "CODE39"
  },
  "contactRequired": false,
  "courierNotes": "string",
  "identificationVerified": true,
  "minimumAge": 9999,
  "pictureRequired": false,
  "pictureURL": "string",
  "signatureRequired": false,
  "signatureURL": "string"
},
"pickupVerificationId": <pickupVerificationId>,
"provider": "string",
"providerCreated": "2024-11-28T13:39:37.058Z",
"providerId": "string",
"providerInstance": "string",
"returnTripId": <returnTripId>,
"status": "CREATED",
"statusHistory": [
  {
    "created": "2024-11-28T13:39:37.058Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:39:37.058Z",
    "updatedBy": "string",
    "id": <id>,
    "newStatus": "CREATED",
    "parentId": <parentId>,
    "priorStatus": "CREATED"
  }
],
"tip": 10000000000000000,
"totalValue": 10000000000000000,
"trackingURL": "string",
"undeliverableInfo": {
  "created": "2024-11-28T13:39:37.058Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:39:37.058Z",
  "updatedBy": "string",
  "id": <id>,
  "moreInfo": {
    "additionalProp1": "string",
    "additionalProp2": "string",
```

```
        "additionalProp3": "string"
      },
      "reason": "string",
      "undeliverableAction": "LEAVE_AT_DOOR"
    },
    "undeliverableInfoId": <undeliverableInfoId>
  }
}
```

Schema — Delivery

For more information about the schema Delivery, see the [Schema — Delivery](#) section.

Response Code: 404

The record could not be found.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Check if a Delivery Exists — HEAD

This section describes the endpoint which checks if a delivery exists.

Method

HEAD

URL

/api/delivery/v5/delivery/{id}

Allowed Scope(s)

delivery-r
delivery-rw

Request

This section describes the request parameters.

Table 4-59 Checks if a Delivery Exists — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer <int64> (query)	The identifier of the delivery.

Responses

This section describes the responses of the Check if Delivery Exists endpoint.

Response Code: 200

The requested record exists.

Response Code: 404

The record could not be found.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Delivery Records — GET

This section describes the Queries for Delivery Records endpoint. This service returns a list of delivery records and the associated delivery details matching the provided criteria.

Method

GET

URL

`/api/delivery/v5/delivery`

Allowed Scope(s)

`delivery-r`
`delivery-rw`

Request

This section describes the request parameters.

Table 4-60 Queries for Delivery Records — Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, that is a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example, {"child.name":"Jo"} would match Joe, John, etc. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example, {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is "@and1",</p>

Table 4-60 (Cont.) Queries for Delivery Records — Request Parameters

Parameter	Required	Data Type	Description
orderBy	NA	string (query)	<p>"@and2", and so on) to ensure that they are uniquely named since any duplicates will be eliminated.</p> <p>A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by "asc" (default) or "desc" to control order direction.</p> <p>When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.</p>
expands	NA	string (query)	<p>A comma-delimited list of relationship paths, in dot-notated path form (that is "child.line" would return the relationship named "child", and its child relationship named "line"), on the resulting model which should be included in the response or "all" if all associated relationships should be included.</p>
limit	NA	integer(\$int32) (query)	<p>Provides a maximum number of records to be returned.</p>
offset	NA	integer(\$int32) (query)	<p>Provides the zero-based value for the position of the first value to be returned.</p>

Responses

This section describes the responses of the Queries for Delivery Records endpoint.

Response Code: 200

The requested records.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-11-28T13:46:35.808Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:46:35.808Z",
    "updatedBy": "string",
```



```
"id": <id>,
"barcodes": [
  {
    "created": "2024-11-28T13:46:35.808Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:46:35.808Z",
    "updatedBy": "string",
    "id": <id>,
    "identifier": "string",
    "parentId": <parentId>,
    "type": "CODE39",
    "value": "string"
  }
],
"cancelReasonCode": "string",
"canceledBy": "string",
"crossReferences": [
  {
    "created": "2024-11-28T13:46:35.808Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:46:35.808Z",
    "updatedBy": "string",
    "id": <id>,
    "key": "string",
    "parentId": <parentId>,
    "value": "string"
  }
],
"currency": "str",
"deliveryCancelReason": "COURIER",
"deliveryId": "string",
"deliveryType": "DELIVERY",
"dropoffAddress": {
  "created": "2024-11-28T13:46:35.808Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:46:35.808Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"dropoffAddressId": <dropoffAddressId>,
"dropoffContact": {
  "created": "2024-11-28T13:46:35.808Z",
  "createdBy": "string",
```

```
"updated": "2024-11-28T13:46:35.808Z",
"updatedBy": "string",
"id": <id>,
"businessName": "string",
"contactReference": "string",
"emailAddress": "string",
"firstName": "string",
"lastName": "string",
"middleName": "string",
"phoneNumber": "string",
"suffix": "string"
},
"dropoffContactId": <dropoffContactId>,
"dropoffDeadline": "2024-11-28T13:46:35.808Z",
"dropoffEta": "2024-11-28T13:46:35.808Z",
"dropoffLocationCode": "string",
"dropoffNotes": "string",
"dropoffReady": "2024-11-28T13:46:35.808Z",
"dropoffVerification": {
  "created": "2024-11-28T13:46:35.808Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:46:35.808Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodeRequired": false,
  "barcodes": {
    "additionalProp1": "CODE39",
    "additionalProp2": "CODE39",
    "additionalProp3": "CODE39"
  },
  "contactRequired": false,
  "courierNotes": "string",
  "identificationVerified": true,
  "minimumAge": 9999,
  "pictureRequired": false,
  "pictureURL": "string",
  "signatureRequired": false,
  "signatureURL": "string"
},
"dropoffVerificationId": <dropoffVerificationId>,
"errorCount": 9999,
"fee": 10000000000000000,
"labelsAttached": false,
"manifest": [
  {
    "created": "2024-11-28T13:46:35.808Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:46:35.808Z",
    "updatedBy": "string",
    "id": <id>,
    "description": "string",
    "height": 9999,
    "identifier": "string",
    "items": [
      {
        "created": "2024-11-28T13:46:35.808Z",
```

```
        "createdBy": "string",
        "updated": "2024-11-28T13:46:35.808Z",
        "updatedBy": "string",
        "id": <id>,
        "description": "string",
        "identifier": "string",
        "parentId": <parentId>,
        "price": 10000000000000000,
        "quantity": 9999
      }
    ],
    "length": 9999,
    "packDimensionCode": "string",
    "parentId": <parentId>,
    "totalAmount": 10000000000000000,
    "weight": 10000000000000000,
    "width": 9999
  }
],
"manifestSize": 9999,
"pickupAddress": {
  "created": "2024-11-28T13:46:35.808Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:46:35.808Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
  "state": "string"
},
"pickupAddressId": <pickupAddressId>,
"pickupContact": {
  "created": "2024-11-28T13:46:35.808Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:46:35.808Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
```

```
"pickupContactId": <pickupContactId>,
"pickupDeadline": "2024-11-28T13:46:35.808Z",
"pickupEta": "2024-11-28T13:46:35.808Z",
"pickupLocationCode": "string",
"pickupNotes": "string",
"pickupReady": "2024-11-28T13:46:35.808Z",
"pickupVerification": {
  "created": "2024-11-28T13:46:35.808Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:46:35.808Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodeRequired": false,
  "barcodes": {
    "additionalProp1": "CODE39",
    "additionalProp2": "CODE39",
    "additionalProp3": "CODE39"
  },
  "contactRequired": false,
  "courierNotes": "string",
  "identificationVerified": true,
  "minimumAge": 9999,
  "pictureRequired": false,
  "pictureURL": "string",
  "signatureRequired": false,
  "signatureURL": "string"
},
"pickupVerificationId": <pickVerificationId>,
"provider": "string",
"providerCreated": "2024-11-28T13:46:35.808Z",
"providerId": "string",
"providerInstance": "string",
"returnTripId": <returnTripId>,
"status": "CREATED",
"statusHistory": [
  {
    "created": "2024-11-28T13:46:35.808Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:46:35.808Z",
    "updatedBy": "string",
    "id": 9223372036854776000,
    "newStatus": "CREATED",
    "parentId": <parentId>,
    "priorStatus": "CREATED"
  }
],
"tip": 10000000000000000,
"totalValue": 10000000000000000,
"trackingURL": "string",
"undeliverableInfo": {
  "created": "2024-11-28T13:46:35.808Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:46:35.808Z",
  "updatedBy": "string",
  "id": <id>,
  "moreInfo": {
```

```
        "additionalProp1": "string",
        "additionalProp2": "string",
        "additionalProp3": "string"
      },
      "reason": "string",
      "undeliverableAction": "LEAVE_AT_DOOR"
    },
    "undeliverableInfoId": <undeliverableInfoId>
  }
]
```

Schema — Delivery

For more information about the schema Delivery, see the [Schema — Delivery](#) section.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Delivery Records/Count — GET

This section describes the Queries for Delivery Records Count endpoint. This service returns a count of records matching the provided criteria.

Method

GET

URL

/api/delivery/v5/delivery/count

Allowed Scope(s)

delivery-r

Request

This section describes the request parameters.

Table 4-61 Queries for a Delivery Records Count — Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, that is a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example, {"child.name":"Jo"} would match Joe, John, and son on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example, {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example, {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is "@and1",</p>

Table 4-61 (Cont.) Queries for a Delivery Records Count — Request Parameters

Parameter	Required	Data Type	Description
			"@and2", and so on) to ensure that they are uniquely named since any duplicates will be eliminated.

Responses

This section describes the responses of the Queries for Delivery Records Count endpoint.

Response Code: 200

The requested records.

The media type is application/json.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Delivery Records by Attributes — GET

This section describes the Queries for Delivery Records by Attributes endpoint. This endpoint returns a list of records matching the provided attributes.

Method

GET

URL

/api/deliver/v5/delivery/byattributes

Allowed Scope(s)

delivery-r

Request

This section describes the request parameters.

Table 4-62 Queries for Delivery Records by Attributes — Request Parameters

Parameter	Required	Data Type	Description
id	NA	integer (\$int64) (query)	ID
cancelReasonCode	NA	string (query)	cancel Reason Code
canceledBy	NA	string (query)	canceled by
currency	NA	string (query)	currency
deliveryCancelReason	NA	string (query)	delivery cancel reason Available values: COURIER, PROVIDER, CUSTOMER
deliveryId	NA	string (query)	delivery ID
deliveryType	NA	string (query)	delivery type; Available values: DELIVERY, RETURN; Default value: DELIVERY
dropoffAddressId	NA	integer (\$int64) (query)	dropoff address ID
dropoffContactId	NA	integer (\$int64) (query)	dropoff contact ID
dropoffDeadline	NA	string (\$date-time) (query)	dropoff deadline
dropoffEta	NA	string (\$date-time) (query)	dropoff Eta
dropoffLocationCode	NA	string (query)	dropoff location code
dropoffNotes	NA	string (query)	dropoff notes
dropoffReady	NA	string (\$date-time) (query)	dropoff ready
dropoffVerificationId	NA	integer (\$int64) (query)	dropoff verification ID
errorCount	NA	integer (\$int32) (query)	error count
fee	NA	number (query)	fee
labelsAttached	NA	boolean (query)	Default value : false Note: Return delivery types have the following limitations with the Uber integration: Labels Attached will always be set to TRUE which indicates that a label will be attached to the package(s).
manifestSize	NA	integer (\$int32) (query)	manifest size
pickupAddressId	NA	integer (\$int64) (query)	pickup address ID
pickupContactId	NA	integer (\$int64) (query)	pickup contact ID
pickupDeadline	NA	string (\$date-time) (query)	pickup deadline
pickupEta	NA	string (\$date-time) (query)	pickup Eta
pickupLocationCode	NA	string (query)	pickup location code
pickupNotes	NA	string (query)	pickup notes
pickupReady	NA	string (\$date-time) (query)	pickup ready
pickupVerificationId	NA	integer (\$int64) (query)	pickup verification ID

Table 4-62 (Cont.) Queries for Delivery Records by Attributes — Request Parameters

Parameter	Required	Data Type	Description
provider	NA	string (query)	provider
providerCreated	NA	string (\$date-time) (query)	provider created
providerId	NA	string (query)	provider ID
providerInstance	NA	string (query)	provider instance
returnTripId		integer(\$int64) (query)	return trip ID
status	NA	string (query)	Available values : CREATED, PENDING, ENROUTE_PICKUP, PICKUP_COMPLETE, ENROUTE_DROPOFF, DROPOFF_COMPLETE, CANCELED, RETURNED, ERROR, ENROUTE_RETURN
tip	NA	number (query)	tip
totalValue	NA	number (query)	total value
trackingURL	NA	string (query)	tracking URL
undeliverableInfoId	NA	integer (\$int64) (query)	undeliverable info ID
orderBy	NA	string (query)	A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by " asc " (default) or " desc " to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.
limit	NA	integer (\$int32) (query)	Provides a maximum number of records to be returned.
offset	NA	integer (\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for Delivery Records by Attributes endpoint.

Response Code: 200

The requested records.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-11-28T13:53:35.285Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:53:35.285Z",
    "updatedBy": "string",
    "id": <id>,
    "barcodes": [
      {
        "created": "2024-11-28T13:53:35.285Z",
        "createdBy": "string",
        "updated": "2024-11-28T13:53:35.285Z",
        "updatedBy": "string",
        "id": <id>,
        "identifier": "string",
        "parentId": <parentId>,
        "type": "CODE39",
        "value": "string"
      }
    ],
    "cancelReasonCode": "string",
    "canceledBy": "string",
    "crossReferences": [
      {
        "created": "2024-11-28T13:53:35.285Z",
        "createdBy": "string",
        "updated": "2024-11-28T13:53:35.285Z",
        "updatedBy": "string",
        "id": <id>,
        "key": "string",
        "parentId": <parentId>,
        "value": "string"
      }
    ],
    "currency": "str",
    "deliveryCancelReason": "COURIER",
    "deliveryId": "string",
    "deliveryType": "DELIVERY",
    "dropoffAddress": {
      "created": "2024-11-28T13:53:35.285Z",
      "createdBy": "string",
      "updated": "2024-11-28T13:53:35.285Z",
      "updatedBy": "string",
      "id": <id>,
```

```
"address1": "string",
"address2": "string",
"address3": "string",
"address4": "string",
"addressType": "RESIDENTIAL",
"apartment": "string",
"city": "string",
"country": "string",
"latitude": 999999999,
"locationCode": "string",
"longitude": 999999999,
"postalCode": "string",
"state": "string"
},
"dropoffAddressId": <dropoffAddressId>,
"dropoffContact": {
  "created": "2024-11-28T13:53:35.285Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:53:35.285Z",
  "updatedBy": "string",
  "id": <id>,
  "businessName": "string",
  "contactReference": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "middleName": "string",
  "phoneNumber": "string",
  "suffix": "string"
},
"dropoffContactId": <dropoffContactId>,
"dropoffDeadline": "2024-11-28T13:53:35.285Z",
"dropoffEta": "2024-11-28T13:53:35.285Z",
"dropoffLocationCode": "string",
"dropoffNotes": "string",
"dropoffReady": "2024-11-28T13:53:35.285Z",
"dropoffVerification": {
  "created": "2024-11-28T13:53:35.285Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:53:35.285Z",
  "updatedBy": "string",
  "id": <id>,
  "barcodeRequired": false,
  "barcodes": {
    "additionalProp1": "CODE39",
    "additionalProp2": "CODE39",
    "additionalProp3": "CODE39"
  },
  "contactRequired": false,
  "courierNotes": "string",
  "identificationVerified": true,
  "minimumAge": 9999,
  "pictureRequired": false,
  "pictureURL": "string",
  "signatureRequired": false,
  "signatureURL": "string"
```

```
},
"dropoffVerificationId": 9223372036854776000,
"errorCount": 9999,
"fee": 10000000000000000,
"labelsAttached": false,
"manifest": [
  {
    "created": "2024-11-28T13:53:35.285Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:53:35.285Z",
    "updatedBy": "string",
    "id": 9223372036854776000,
    "description": "string",
    "height": 9999,
    "identifier": "string",
    "items": [
      {
        "created": "2024-11-28T13:53:35.285Z",
        "createdBy": "string",
        "updated": "2024-11-28T13:53:35.285Z",
        "updatedBy": "string",
        "id": <id>,
        "description": "string",
        "identifier": "string",
        "parentId": 9223372036854776000,
        "price": 10000000000000000,
        "quantity": 9999
      }
    ],
    "length": 9999,
    "packDimensionCode": "string",
    "parentId": 9223372036854776000,
    "totalAmount": 10000000000000000,
    "weight": 10000000000000000,
    "width": 9999
  }
],
"manifestSize": 9999,
"pickupAddress": {
  "created": "2024-11-28T13:53:35.285Z",
  "createdBy": "string",
  "updated": "2024-11-28T13:53:35.285Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "latitude": 999999999,
  "locationCode": "string",
  "longitude": 999999999,
  "postalCode": "string",
```

```
    "state": "string"
  },
  "pickupAddressId": <pickupAddressId>,
  "pickupContact": {
    "created": "2024-11-28T13:53:35.285Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:53:35.285Z",
    "updatedBy": "string",
    "id": <id>,
    "businessName": "string",
    "contactReference": "string",
    "emailAddress": "string",
    "firstName": "string",
    "lastName": "string",
    "middleName": "string",
    "phoneNumber": "string",
    "suffix": "string"
  },
  "pickupContactId": <pickupcontactId>,
  "pickupDeadline": "2024-11-28T13:53:35.285Z",
  "pickupEta": "2024-11-28T13:53:35.285Z",
  "pickupLocationCode": "string",
  "pickupNotes": "string",
  "pickupReady": "2024-11-28T13:53:35.285Z",
  "pickupVerification": {
    "created": "2024-11-28T13:53:35.285Z",
    "createdBy": "string",
    "updated": "2024-11-28T13:53:35.285Z",
    "updatedBy": "string",
    "id": <id>,
    "barcodeRequired": false,
    "barcodes": {
      "additionalProp1": "CODE39",
      "additionalProp2": "CODE39",
      "additionalProp3": "CODE39"
    },
    "contactRequired": false,
    "courierNotes": "string",
    "identificationVerified": true,
    "minimumAge": 9999,
    "pictureRequired": false,
    "pictureURL": "string",
    "signatureRequired": false,
    "signatureURL": "string"
  },
  "pickupVerificationId": <pickupVerificationId>,
  "provider": "string",
  "providerCreated": "2024-11-28T13:53:35.285Z",
  "providerId": "string",
  "providerInstance": "string",
  "returnTripId": <returnTripId>,
  "status": "CREATED",
  "statusHistory": [
    {
      "created": "2024-11-28T13:53:35.285Z",
      "createdBy": "string",
```

```

        "updated": "2024-11-28T13:53:35.285Z",
        "updatedBy": "string",
        "id": <id>,
        "newStatus": "CREATED",
        "parentId": <parentId>,
        "priorStatus": "CREATED"
      }
    ],
    "tip": 10000000000000000,
    "totalValue": 10000000000000000,
    "trackingURL": "string",
    "undeliverableInfo": {
      "created": "2024-11-28T13:53:35.285Z",
      "createdBy": "string",
      "updated": "2024-11-28T13:53:35.285Z",
      "updatedBy": "string",
      "id": <id>,
      "moreInfo": {
        "additionalProp1": "string",
        "additionalProp2": "string",
        "additionalProp3": "string"
      },
      "reason": "string",
      "undeliverableAction": "LEAVE_AT_DOOR"
    },
    "undeliverableInfoId": <undeliverableInfoId>
  }
]

```

Schema — Delivery

For more information about the schema Delivery, see the [Schema — Delivery](#) section.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Metrics Services

This section describes the Metrics Services.

Metrics — GET

This section describes the Metrics endpoint. This service retrieves delivery metrics for created, completed, canceled, and returned deliveries based on specified parameters such as delivery type, start and end date, and time zone offset.

Method

GET

URL

/api/delivey/v5/metric/delivery

Allowed Scope(s)

delivery-r

Request

This section describes the request parameters.

Table 4-63 Metrics — Request Parameters

Parameter	Required	Data Type	Description
providerInstance	NA	string (query)	The provider instance of the delivery.
type	NA	string (query)	The delivery type of metric data. Available values: DELIVERY, RETURN Note: If both values are required, leave the type value blank.
dateRange	NA	string (query)	The date range of delivery metric data. dateRange takes precedence and ignores startDate and endDate when specified. Available values: YDY, TDY, MTD, YTD
offset	NA	string (query)	The time-zone offset of dateRange in the form of (+/-)HH:MI between -18:00 and +18:00 (inclusive).
startDateTime	NA	string (\$date-time) (query)	The start datetime of delivery metric data when dateRange is not specified.
endDateTime	NA	string (\$date-time) (query)	The end datetime of delivery metric data when dateRange is not specified.
results	Yes	string (query)	The summary type that determines how data will be aggregated when returned. Available values: DETAIL, DAILY, SINGLE

Responses

This section describes the responses of the Metrics endpoint.

Response Code: 200

The result object.

The media type is application/json.

Example Value

```
[
  {
    "intervalEnd": "2024-07-24T16:18:49.626Z",
    "intervalStart": "2024-07-24T16:18:49.626Z",
    "metrics": [
      {
        "metricType": "DELIVERIES_CREATED",
        "value": 2147483647
      }
    ]
  }
]
```

Schema — Metric Response

The following table describes the Metric Response object.

Table 4-64 Metric Response — Object

Element Name	Required	Data Type	Description
intervalEnd	Yes	string(\$date-time)	The end time of metric data.
intervalStart	Yes	string(\$date-time)	The start time of metric data.
metrics	NA	object	A collection of metric items for a given period.

The following table describes a container for predefined metrics.

Table 4-65 Metric — Object

Element Name	Required	Data Type	Description
metricType	NA	string	A type of metric. Enum: [DELIVERIES_CREATED, DELIVERIES_COMPLETED, DELIVERIES_CANCELED, DELIVERIES_RETURNED, RETURNS_CREATED, RETURNS_COMPLETED, RETURNS_CANCELED, RETURNS_RETURNED]
value	NA	integer (\$int32)	A count of metric.

Response Code: 400

The provided input is valid.

The media type is application/json.

Example

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

For details about the Mapper Error Data schema, see the [Schema — Mapper Error Data](#) section.

Response Code: 500

The request failed.

Location Metrics — GET

This section describes the Location Metrics endpoint. This service retrieves delivery metrics, for a specific location, for created, completed, canceled, and returned deliveries based on specified parameters such as delivery type, start and end date, and time zone offset.

Method

GET

URL

```
/api/delivery/v5/metric/location/{locationCode}
```

Allowed Scope(s)

delivery-r

Request

This section describes the request parameters.

Table 4-66 Location Metrics — Request Parameters

Parameter	Required	Data Type	Description
providerInstance	NA	string (query)	The provider instance of the delivery.
type	NA	string (query)	The delivery type of metric data. All types will be returned if left empty. Available values : DELIVERY, RETURN
dateRange	NA	string (query)	The date range of delivery metric data. dateRange takes precedence and ignores startDate and endDate when specified. Available values : YDY, TDY, MTD, YTD
offset	NA	string (query)	The time-zone offset of dateRange in the form of (+/-)HH:MI between -18:00 and +18:00 (inclusive).
startDateTime	NA	string(\$date-time) (query)	The start datetime of delivery metric data when dateRange is not specified.
endDateTime	NA	string(\$date-time) (query)	The end datetime of delivery metric data when dateRange is not specified.
results	Yes	string (query)	The summary type that determines how data will be aggregated when returned. Available values : DETAIL, DAILY, SINGLE
locationCode	Yes	string (path)	The location code of metric data. A location code that matches a pickup location code will fall under Outbound Metrics data and a location code that matches a dropoff location code will fall under Inbound Metrics data.

Responses

This section describes the responses of the Metrics endpoint.

Response Code: 200

The result object.

The media type is application/json.

Example Value

```
[
  {
    "intervalEnd": "2024-07-24T16:19:45.792Z",
    "intervalStart": "2024-07-24T16:19:45.792Z",
    "locationCode": "string",
    "metricsInbound": [
```

```

    {
      "metricType": "DELIVERIES_CREATED",
      "value": 2147483647
    }
  ],
  "metricsOutbound": [
    {
      "metricType": "DELIVERIES_CREATED",
      "value": 2147483647
    }
  ]
}
]

```

Schema — Location Metric Response

The following table describes the Location Metric Response object. A container for a metric response

Table 4-67 Location Metric Response — Object

Element Name	Required	Data Type	Description
intervalEnd	Yes	string(\$date-time)	The end time of metric data.
intervalStart	Yes	string(\$date-time)	The start time of metric data.
locationCode	NA	string	The location code of metric data where inbound (dropoff location) and outbound (pickup location) metrics are based.
metricsInbound	NA	object	A collection of metric inbound items for a given period if a location code is provided and if the location code matches a dropoff location code.
metricsOutbound	NA	object	A collection of metric outbound items for a given period if a location code is provided and if the location code matches a pickup location code.

The following table describes a container for predefined metrics.

Table 4-68 Metric — Object

Element Name	Required	Data Type	Description
metricType	NA	string	A type of metric. Enum: [DELIVERIES_CREATED, DELIVERIES_COMPLETED, DELIVERIES_CANCELED, DELIVERIES_RETURNED, RETURNS_CREATED, RETURNS_COMPLETED, RETURNS_CANCELED, RETURNS_RETURNED]
value	NA	integer (\$int32)	A count of metric.

Response Code: 400

The provided input is valid.

The media type is application/json.

Example

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

For details about the Mapper Error Data schema, see the [Schema — Mapper Error Data](#) section.

Response Code: 500

The request failed.

Webhook Credential Services

This section describes the Webhook Credential Services.

Add Credential — POST

This section describes the AddCredential API. This API creates a new credential.

Method

POST

URL

```
/api/delivery/v5/webhook/credential
```

Allowed Scope(s)

```
webhook-rw
```

Request

There are no request parameters.

The request body is application/json.

A request indicating the parameters of the credential.

Example Value

```
{
  "clientId": "string",
  "clientSecret": "string",
  "credentialId": "string",
  "description": "string",
  "id": <id>,
  "scope": "string",
  "url": "string"
}
```

Schema — Credential Request

The following table describes a request to create/update a credential used when accessing an external service.

Table 4-69 CredentialRequest — Object

Element Name	Required	Data Type	Description
clientId	yes	string(256)	The client ID for the credential.
clientSecret	yes	string(256)	The client secret for the credential.
credentialId	yes	string(32)	The credential ID.
description	NA	string(256)	The credential description, if one has been defined.
id	NA	integer(\$int64)	ID
scope	NA	string	The scope for the credential.
url	yes	string	The OAuth2 URL for the credential.

Responses

This section describes the responses of the Add Credential endpoint.

Response Code: 200

The request was processed successfully.

The media type is application/json.

Example Value

```
{
  "created": "2024-01-19T10:11:38.601Z",
  "createdBy": "string",
  "credentialId": "string",
  "description": "string",
  "id": <id>,
  "updated": "2024-01-19T10:11:38.601Z",
  "updatedBy": "string"
}
```

Schema — Credential Response

The table below describes a credential used when accessing an external service..

Table 4-70 Credential Response — Object

Element Name	Required	Data Type	Description
created	yes	string (\$date-time)	The timestamp at which the credential was created.
created By	yes	string	The user that created the credential.
credentialId	yes	string(32)	The credential ID.
description	NA	string(256)	The credential description, if one has been defined.
id	NA	integer(\$int64)	ID
updated	NA	string(\$date-time)	The timestamp at which the credential was last updated, or null if it has never been updated.
updatedBy	NA	string	The user that last updated the credential, or null if it has never been updated.

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 4-71 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 4-72 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.

Table 4-72 (Cont.) More Info — Object

Element Name	Required	Data Type	Description
description	NA	string	An unlocalizable description describing the error.

Response Code: 409

The provided credentialId already exists.

Response Code: 500

A server error prevented successful processing of the request.

Get Credentials — GET

This section describes the GetCredentials API. This API obtains a set of known credentials.

Method

GET

URL

/api/delivery/v5/webhook/credential

Allowed Scope(s)

webhook-rw

Request

There are no request parameters.

Responses

This section describes the responses of the GetCredentials API.

Response Code: 200

A set of known credentials.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-01-19T10:20:01.887Z",
    "createdBy": "string",
    "credentialId": "string",
    "description": "string",
    "id": <id>,
    "updated": "2024-01-19T10:20:01.887Z",
    "updatedBy": "string"
  }
]
```

Schema — Credential Response

The table below describes a credential used when accessing an external service..

Table 4-73 Credential Response — Object

Element Name	Required	Data Type	Description
created	yes	string (\$date-time)	The timestamp at which the credential was created.
created By	yes	string	The user that created the credential.
credentialId	yes	string(32)	The credential ID.
description	NA	string(256)	The credential description, if one has been defined.
id	NA	integer(\$int64)	ID
updated	NA	string(\$date-time)	The timestamp at which the credential was last updated, or null if it has never been updated.
updatedBy	NA	string	The user that last updated the credential, or null if it has never been updated.

Response Code: 500

A server error prevented successful processing of the request.

Get Credential — GET

This section describes the GetCredential API. This API obtains a credential.

Method

GET

URL

```
/api/delivery/v5/webhook/credential/{id}
```

Allowed Scope(s)

```
webhook-rw
```

Request

This section describes the request parameters.

Table 4-74 Get Credential — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64)	ID

Responses

This section describes the responses of the GetCredential API.

Response Code: 200

The requested credential.

The media type is application/json.

Example Value

```
{
  "created": "2024-05-22T11:55:10.379Z",
  "createdBy": "string",
  "credentialId": "string",
  "description": "string",
  "id": <id>,
  "updated": "2024-05-22T11:55:10.379Z",
  "updatedBy": "string"
}
```

Schema — Credential Response

The table below describes a credential used when accessing an external service..

Table 4-75 Credential Response — Object

Element Name	Required	Data Type	Description
created	yes	string (\$date-time)	The timestamp at which the credential was created.
created By	yes	string	The user that created the credential.
credentialId	yes	string(32)	The credential ID.
description	NA	string(256)	The credential description, if one has been defined.
id	NA	integer(\$int64)	ID
updated	NA	string(\$date-time)	The timestamp at which the credential was last updated, or null if it has never been updated.
updatedBy	NA	string	The user that last updated the credential, or null if it has never been updated.

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 4-76 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 4-77 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Response Code: 404

The requested credential could not be found.

Response Code: 500

A server error prevented successful processing of the request.

Update Credential — PUT

This section describes the UpdateCredential API. This API updates a credential.

Method

PUT

URL

```
/api/delivery/v5/webhook/credential/{id}
```

Allowed Scope(s)

```
webhook-rw
```

Request

This table describes the request parameters of the DeleteCredential API.

The request body is application/json.

A request indicating the parameters of the credential.

Table 4-78 UpdateCredential — Request Parameters

Parameter	Required	Data Type	Description
id	yes	integer(\$int64) (path)	The credential id to be updated.

Example Value

```
{
  "clientId": "string",
  "clientSecret": "string",
  "credentialId": "string",
  "description": "string",
  "id": <id>,
  "scope": "string",
  "url": "string"
}
```

Schema — Credential Request

The following table describes a request to create/update a credential used when accessing an external service.

Table 4-79 CredentialRequest — Object

Element Name	Required	Data Type	Description
clientId	yes	string(256)	The client ID for the credential.
clientSecret	yes	string(256)	The client secret for the credential.
credentialId	yes	string(32)	The credential ID.
description	NA	string(256)	The credential description, if one has been defined.

Table 4-79 (Cont.) CredentialRequest — Object

Element Name	Required	Data Type	Description
id	NA	integer(\$int64)	ID
scope	NA	string	The scope for the credential.
url	yes	string	The OAuth2 URL for the credential.

Responses

This section describes the responses of the UpdateCredential API.

Response Code: 200

The request was processed successfully.

The media type is application/json.

Example Value

```
{
  "created": "2024-01-19T10:27:47.428Z",
  "createdBy": "string",
  "credentialId": "string",
  "description": "string",
  "id": <id>,
  "updated": "2024-01-19T10:27:47.428Z",
  "updatedBy": "string"
}
```

Schema — Credential Response

The table below describes a credential used when accessing an external service..

Table 4-80 Credential Response — Object

Element Name	Required	Data Type	Description
created	yes	string (\$date-time)	The timestamp at which the credential was created.
created By	yes	string	The user that created the credential.
credentialId	yes	string(32)	The credential ID.
description	NA	string(256)	The credential description, if one has been defined.
id	NA	integer(\$int64)	ID
updated	NA	string(\$date-time)	The timestamp at which the credential was last updated, or null if it has never been updated.

Table 4-80 (Cont.) Credential Response — Object

Element Name	Required	Data Type	Description
updatedBy	NA	string	The user that last updated the credential, or null if it has never been updated.

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 4-81 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error

Table 4-81 (Cont.) Mapper Error Data — Object

Element Name	Required	Data Type	Description
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 4-82 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Response Code: 404

The requested credential could not be found.

Response Code: 500

A server error prevented successful processing of the request.

Delete Credential — DELETE

This section describes the DeleteCredential API. This API deletes a credential.

Method

DELETE

URL

/api/delivery/v5/webhook/credential/{id}

Allowed Scope(s)

webhook-rw

Request

This table describes the request parameters of the DeleteCredential API.

Table 4-83 DeleteCredential — Request Parameters

Parameter	Required	Data Type	Description
id	yes	integer(\$int64) (path)	The credential id to be deleted.

Responses

This section describes the responses of the DeleteCredential API.

Response Code: 204

The credential was successfully found and deleted.

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 4-84 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 4-85 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Response Code: 404

The requested credential could not be found.

Response Code: 409

The provided credential ID is in use and cannot be deleted.

Response Code: 500

A server error prevented successful processing of the request.

Status Notification Webhook Services

This section describes the Status Notification Webhook Services.

Collect and Receive can notify external applications about updates to the statuses of deliveries by webhooks. Webhook notifications occur whenever the status of a delivery changes, and

allows interested applications to subscribe to notifications in near-real-time. Note the following details:

- Notifications are sent via REST, authenticated using OAuth 2.0, and may include up to 100 notifications per batch.
- Consuming applications must acknowledge the receipt of notifications by responding as per the requirements of the API.
- Delivery attempts will be made for up to 48 hours if notifications cannot be transmitted successfully.
 - Any notification that could not reach its destination, or was not acknowledged, is treated as unsuccessful.
 - Each retry attempt will be attempted with increasing delay. The first retry will occur after 30 seconds, and each subsequent attempt increases the delay period by 4x.

The webhook API, which would be implemented by an application interested in these notifications, looks as following:

```
{
  "openapi": "3.0.1",
  "info": {
    "title": "Oracle Collect and Receive Delivery Notification",
    "version": "2.0.0",
    "description": "A delivery notification API for Oracle Collect and
Receive"
  },
  "paths": {
    "/notify": {
      "post": {
        "summary": "notify",
        "description": "Send delivery status notifications.",
        "operationId": "notify",
        "tags": [
          "WebhookV2"
        ],
        "requestBody": {
          "description": "Delivery with pickup or dropOff delayed
notifications.",
          "content": {
            "application/json": {
              "schema": {
                "type": "array",
                "items": {
                  "$ref": "#/components/schemas/
NotificationV2"
                }
              }
            }
          }
        },
        "responses": {
          "200": {
            "description": "Successfully received delivery IDs.
This array must contain the unique identifiers of deliveries that have been
successfully received from the request body.",
            "content": {
```

```

        "application/json": {
            "schema": {
                "type": "array",
                "items": {
                    "maximum": 9223372036854776000,
                    "minimum": -9223372036854776000,
                    "type": "integer",
                    "format": "int64",
                    "description": "The unique identifier
of the delivery."
                }
            }
        }
    },
    "components": {
        "schemas": {
            "NotificationV2": {
                "type": "object",
                "properties": {
                    "id": {
                        "maximum": 9223372036854776000,
                        "minimum": -9223372036854776000,
                        "type": "integer",
                        "format": "int64",
                        "description": "The unique identifier of the
delivery."
                    },
                    "deliveryId": {
                        "maxLength": 128,
                        "minLength": 0,
                        "type": "string",
                        "description": "The primary reference ID from the
calling system uniquely identifying this delivery."
                    },
                    "deliveryAttributes": {
                        "description": "The delivery identifier attributes.",
                        "$ref": "#/components/schemas/DeliveryAttributes"
                    },
                    "status": {
                        "description": "The last known status of the
delivery.",
                        "$ref": "#/components/schemas/DeliveryStatus"
                    },
                    "updated": {
                        "type": "string",
                        "description": "The timestamp at which the record was
last updated.",
                        "format": "date-time"
                    }
                }
            }
        }
    }
}

```

```

    "DeliveryStatus": {
      "type": "string",
      "description": "The last known status of the delivery.",
      "enum": [
        "CREATED",
        "PENDING",
        "ENROUTE_PICKUP",
        "PICKUP_COMPLETE",
        "ENROUTE_DROPOFF",
        "DROPOFF_COMPLETE",
        "CANCELED",
        "RETURNED",
        "ERROR",
        "ENROUTE_RETURN"
      ]
    },
    "DeliveryAttributes": {
      "type": "array",
      "description": "The delivery identifier attributes.",
      "items": {
        "type": "string",
        "description": "The enumeration of attributes.",
        "enum": ["PICKUP_DELAYED", "DROPOFF_DELAYED"]
      }
    }
  }
}

```

Leveraging the Status Notification Webhook for Delivery Communications

The Collect and Receive Status Notification Webhook can be leveraged and used to facilitate and trigger communication to the sending and/or receiving parties of a delivery. Leveraging the webhook to send communications is beneficial specifically for external clients, businesses, or users, that are not part of your organization and do not have access to the Collect and Receive UI, but need updates about the progress of a delivery.

It is strongly recommended that the business system(s) that are receiving the Status Notification webhooks use the notifications to get the delivery details, create package labels, and communicate them to the contact at the pickup location. As part of the delivery process, labels must be attached to all packages for the courier to scan during transport. When the contact at the pickup location does not have access to the Collect and Receive UI, they will not be able to retrieve the package labels.

Sending communications externally can be accomplished by doing the following on receipt of the Status Notification Webhook:

1. The receiving system can call the Collect and Receive `getStatus` API to retrieve all of the details about the delivery
2. Trigger, via your business' own communication tool (i.e. Email Service Provider), an external notification (i.e. email, SMS, push notification) about the status of an order and the ongoing delivery process. (Examples: Delivery Scheduled Successfully, Delivery is delayed, Packages are out for delivery, and so on)

3. If the status in the webhook notification was pending, use the delivery details to create and send package label(s) to the contact at the pickup location. Package labels and barcodes must be affixed to each package being sent in a delivery prior to courier pickup. When creating a package label, consider including the following:
 - a. Delivery ID - (Required)
 - b. Manifest Identifier - Printed as a QR code (Required)
 - c. Package X of Y
 - d. Pickup Contact and Address Details
 - e. Dropoff Contact and Address Details

The following are sample scenarios that articulate the benefits of leveraging the status notifications to send external communications:

Scenario 1: A business user creates an inbound delivery via the Collect and Receive User Interface, for packages to be picked up from a non-Collect and Receive user.

When not leveraging the Status Notification Webhook, the non-Collect and Receive user will:	When leveraging the Status Notification Webhook, the non-Collect and Receive user can:
<ul style="list-style-type: none"> Not receive a communication confirming the pickup and delivery details Not be aware they need to print and affix package labels Not be sent any labels for their packages Not know when the courier is enroute to pickup the packages 	<ul style="list-style-type: none"> Receive a communication confirming the pickup and delivery details. Receive communication that includes the package labels with instructions on how/where to affix them. Receive added communication when the courier is enroute to pickup the packages Receive communication if a delivery is delayed.

Scenario 2: A business user creates an outbound delivery via the Collect and Receive User Interface, for packages to be sent to a non-Collect and Receive user.

When not leveraging the Status Notification Webhook, the non-Collect and Receive user will:	When leveraging the Status Notification Webhook, the non-Collect and Receive user can:
<ul style="list-style-type: none"> Not receive a communication confirming the pickup and delivery details. This may include instructions such as having an ID ready for Age Verification or the ability to track the delivery/courier. Not receive on-going communications when as the delivery progresses in status through completion. 	<ul style="list-style-type: none"> Receive communications throughout the entire delivery lifecycle. Receive communication if a delivery is delayed.

Add a New Webhook — POST

This section describes the Add a New Webhook API.

Method

POST

URL

/api/delivery/v5/webhook

Allowed Scope(s)

webhook-rw

Request

This section describes the request parameters.

The request body is application/json.

The ModelField is null.

Table 4-86 Add a New Webhook — Request Parameters

Parameter	Required	Data Type	Description
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is "child.line" would return the relationship named "child", and its child relationship named "line"), on the resulting model which should be included in the response or "all" if all associated relationships should be included.

Example Value

```
{
  "created": "2023-12-14T09:51:05.370Z",
  "createdBy": "string",
  "updated": "2023-12-14T09:51:05.370Z",
  "updatedBy": "string",
  "id": <id>,
  "credentialId": "string",
  "description": "string",
  "url": "string",
  "webhookId": "string",
  "webhookType": "STATUS_NOTIFICATION_V1"
}
```

Schema — Webhook

The following table describes the elements of a webhook used to send delivery status notifications to an external service.

Table 4-87 Webhook — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
credentialId	Yes	string(32)	The credential ID.
description	NA	string(256)	The webhook description, if one has been defined.
url	Yes	string(1024)	The URL for the webhook.
webhookId	Yes	string(32)	The webhook ID. Webhook IDs must contain only letters, numbers, and underscores.
webhookType	Yes	string	Enum: STATUS_NOTIFICATION_V1, STATUS_NOTIFICATION_V2

Responses

This section describes the responses of the Add a New Webhook API.

Reason Code: 200

The new record.

The media type is application/json.

Example Value

```
{
  "created": "2023-12-14T09:51:05.435Z",
  "createdBy": "string",
  "updated": "2023-12-14T09:51:05.435Z",
  "updatedBy": "string",
  "id": <id>,
  "credentialId": "string",
  "description": "string",
  "url": "string",
  "webhookId": "string",
  "webhookType": "STATUS_NOTIFICATION_V1"
}
```

Schema — Webhook

The following table describes the elements of a webhook used to send delivery status notifications to an external service.

Table 4-88 Webhook — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
credentialId	Yes	string(32)	The credential ID.
description	NA	string(256)	The webhook description, if one has been defined.
url	Yes	string(1024)	The URL for the webhook.
webhookId	Yes	string(32)	The webhook ID. Webhook IDs must contain only letters, numbers, and underscores.
webhookType	Yes	string	Enum: STATUS_NOTIFICATION_V1, STATUS_NOTIFICATION_V2

Reason Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 4-89 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 4-90 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 409

The record already exists or a related entity was not found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Webhook Records — GET

This section describes the Queries for Webhook Records API. This API returns a list of records matching the provided criteria.

Method

GET

URL

`/api/delivery/v5/webhook`

Allowed Scope(s)

`webhook-rw`

Request

This section describes the request parameters.

Table 4-91 Queries for Webhook Records— Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, that is a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example {"child.name":"Jo"} would match Joe, John, and so on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is "@and1",</p>

Table 4-91 (Cont.) Queries for Webhook Records— Request Parameters

Parameter	Required	Data Type	Description
orderBy	NA	string (query)	"@and2", and so on) to ensure that they are uniquely named since any duplicates will be eliminated. A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by "asc" (default) or "desc" to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is "child.line" would return the relationship named "child", and its child relationship named "line"), on the resulting model which should be included in the response or "all" if all associated relationships should be included.
limit	NA	integer(\$int32) (query)	Provides a maximum number of records to be returned.
offset	NA	integer(\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for Webhook Records API.

Response Code: 200

The requested records.

The media type is application/json.

Example Value

```
[
  {
    "created": "2023-12-14T13:29:34.113Z",
    "createdBy": "string",
    "updated": "2023-12-14T13:29:34.113Z",
    "updatedBy": "string",
```

```
    "id": <id>,  
    "credentialId": "string",  
    "description": "string",  
    "url": "string",  
    "webhookId": "string",  
    "webhookType": "STATUS_NOTIFICATION_V1"  
  }  
]
```

Schema — Webhook

The following table describes the elements of a webhook used to send delivery status notifications to an external service.

Table 4-92 Webhook — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
credentialId	Yes	string(32)	The credential ID.
description	NA	string(256)	The webhook description, if one has been defined.
url	Yes	string(1024)	The URL for the webhook.
webhookId	Yes	string(32)	The webhook ID. Webhook IDs must contain only letters, numbers, and underscores.
webhookType	Yes	string	Enum: STATUS_NOTIFICATION_V1, STATUS_NOTIFICATION_V2

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Performs a Bulk Action using a Collection of Webhook — POST

This section describes the Performs a Bulk Action using a Collection of Webhook API. This API returns a list of records to be loaded.

Method

POST

URL

```
/api/delivery/v5/webhook/bulk
```

Allowed Scope(s)

```
webhook-rw
```

Request

This section describes the request parameters.

The request body is application/json.

Table 4-93 Performs a Bulk Action using a Collection of Webhook — Request Parameters

Parameter	Required	Data Type	Description
action	NA	string (query)	Provides an action to be used performing the bulk operation. If not specified, the default of ADD_OR_UPDATE is used. Available values: ADD, ADD_OR_UPDATE, DELETE, UPDATE

Example Value

```
[
  {
    "created": "2023-12-14T13:57:12.105Z",
    "createdBy": "string",
    "updated": "2023-12-14T13:57:12.105Z",
    "updatedBy": "string",
    "id": <id>,
    "credentialId": "string",
    "description": "string",
    "url": "string",
    "webhookId": "string",
    "webhookType": "STATUS_NOTIFICATION_V1"
  }
]
```

Schema — Webhook

The following table describes the elements of a webhook used to send delivery status notifications to an external service.

Table 4-94 Webhook — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
credentialId	Yes	string(32)	The credential ID.
description	NA	string(256)	The webhook description, if one has been defined.
url	Yes	string(1024)	The URL for the webhook.
webhookId	Yes	string(32)	The webhook ID. Webhook IDs must contain only letters, numbers, and underscores.
webhookType	Yes	string	Enum: STATUS_NOTIFICATION_V1, STATUS_NOTIFICATION_V2

Responses

This section describes the responses of the Performs a Bulk Action using a Collection of Webhook API.

Response Code: 200

The result of the loading operation.

The media type is application/json.

Example Value

```
{
  "errorRecords": [
    {
      "index": 0,
      "error": "CONFLICT"
    }
  ],
  "successCount": 0
}
```

Schema — Bulk Operation Result

The following table describes the elements of a Bulk Operation, describing the result of a bulk operation.

Table 4-95 Bulk Operation Result — Object

Element Name	Required	Data Type	Description
errorRecords	NA	object	A list of ErrorRecord objects describing any records which were not successfully processed.
successCount	NA	integer(\$int32)	The count of the records which were successfully processed.

The following table describes an object describing an error which occurred during a bulk operation.

Table 4-96 Error Record— Object

Element Name	Required	Data Type	Description
index	NA	integer(\$int32)	The 0-based index of the associated record which was not processed.
error	NA	string	<p>An enumeration describing the result of a bulk action which was not successfully completed.</p> <ul style="list-style-type: none">• CONFLICT - The record was provided with criteria which conflicts with another record.• DOES_NOT_EXIST - The record was provided with an action which required it to exist, but the record was not found.• ALREADY_EXISTS - The record was provided with an action which required it not to exist, but the record was found.• RELATED_DOES_NOT_EXIST - The record was provided without a required related relationship.• INVALID - The record was invalid. Cases for this include:<ul style="list-style-type: none">– The record was provided as an ADD but included an ID.– The record was provided as a DELETE or UPDATE but lacked an ID.– The record was provided with an illegal/invalid field value.• UNKNOWN - Some other unexpected exception occurred. <p>Enum: [CONFLICT, DOES_NOT_EXIST, ALREADY_EXISTS, RELATED_DOES_NOT_EXIST, INVALID, UNKNOWN]</p>

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

For more information about the Mapper Error Data schema, see the [Schema — Mapper Error Data](#) section.

Reason Code: 413

Too many records were provided.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Webhook Records/Count — GET

This section describes the Queries for Webhook Records/Count API. This API returns a count of records matching the provided criteria.

Method

GET

URL

/api/delivery/v5/webhook/count

Allowed Scope(s)

`webhook-rw`

Request

This section describes the request parameters.

Table 4-97 Queries for Webhook Records/Count — Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, i.e. a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example {"child.name":"Jo"} would match Joe, John, and so on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is "@and1",</p>

Table 4-97 (Cont.) Queries for Webhook Records/Count — Request Parameters

Parameter	Required	Data Type	Description
			"@and2", and so on) to ensure that they are uniquely named since any duplicates will be eliminated.

Responses

This section describes the responses of the Queries for Webhook Records/Count API.

Response Code: 200

The count of requested records.

The media type is application/json.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Get a Webhook — GET

This section describes the Get a Webhook API.

Method

GET

URL

/api/delivery/v5/webhook/{id}

Allowed Scope(s)

webhook-rw

Request

This section describes the request parameters.

Table 4-98 Get a Webhook— Request Parameters

Parameter	Required	Data Type	Description
id	yes	integer(\$int64) (path)	ID
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Responses

This section describes responses of the Get a Webhook API.

Response Code: 200

The requested record.

The media type is applicaton/json.

Example Value

```
{
  "created": "2024-06-05T08:47:32.184Z",
  "createdBy": "string",
  "updated": "2024-06-05T08:47:32.184Z",
  "updatedBy": "string",
  "id": <id>,
  "credentialId": "string",
  "description": "string",
  "url": "string",
  "webhookId": "string",
  "webhookType": "STATUS_NOTIFICATION_V1"
}
```

Schema — Webhook

The following table describes the elements of a webhook used to send delivery status notifications to an external service.

Table 4-99 Webhook — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.

Table 4-99 (Cont.) Webhook — Object

Element Name	Required	Data Type	Description
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
credentialId	Yes	string(32)	The credential ID.
description	NA	string(256)	The webhook description, if one has been defined.
url	Yes	string(1024)	The URL for the webhook.
webhookId	Yes	string(32)	The webhook ID. Webhook IDs must contain only letters, numbers, and underscores.
webhookType	Yes	string	Enum: STATUS_NOTIFICATION_V1, STATUS_NOTIFICATION_V2

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Update a Webhook — PUT

This section describes the Update a Webhook API.

Method

PUT

URL

/api/delivery/v5/webhook/{id}

Allowed Scope(s)

webhook-rw

Request

This section describes the request parameters.

The request body is application/json.

Table 4-100 Update a Webhook— Request Parameters

Parameter	Required	Data Type	Description
id	yes	integer(\$int64) (path)	ID
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Example Value

```
{
  "created": "2023-12-15T11:08:18.908Z",
  "createdBy": "string",
  "updated": "2023-12-15T11:08:18.908Z",
  "updatedBy": "string",
  "id": <id>,
  "credentialId": "string",
  "description": "string",
  "url": "string",
  "webhookId": "string",
  "webhookType": "STATUS_NOTIFICATION_V1"
}
```

Schema — Webhook

The following table describes the elements of a webhook used to send delivery status notifications to an external service.

Table 4-101 Webhook — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
credentialId	Yes	string(32)	The credential ID.

Table 4-101 (Cont.) Webhook — Object

Element Name	Required	Data Type	Description
description	NA	string(256)	The webhook description, if one has been defined.
url	Yes	string(1024)	The URL for the webhook.
webhookId	Yes	string(32)	The webhook ID. Webhook IDs must contain only letters, numbers, and underscores.
webhookType	Yes	string	Enum: STATUS_NOTIFICATION_V1, STATUS_NOTIFICATION_V2

Responses

This section describes the responses of the Update a Webhook API.

Reason Code: 200

The updated record.

The media type application/json.

Example Value

```
{
  "created": "2023-12-15T11:08:18.966Z",
  "createdBy": "string",
  "updated": "2023-12-15T11:08:18.966Z",
  "updatedBy": "string",
  "id": <id>,
  "credentialId": "string",
  "description": "string",
  "url": "string",
  "webhookId": "string",
  "webhookType": "STATUS_NOTIFICATION_V1"
}
```

Schema — Webhook

The following table describes the elements of a webhook used to send delivery status notifications to an external service.

Table 4-102 Webhook — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.

Table 4-102 (Cont.) Webhook — Object

Element Name	Required	Data Type	Description
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
credentialId	Yes	string(32)	The credential ID.
description	NA	string(256)	The webhook description, if one has been defined.
url	Yes	string(1024)	The URL for the webhook.
webhookId	Yes	string(32)	The webhook ID. Webhook IDs must contain only letters, numbers, and underscores.
webhookType	Yes	string	Enum: STATUS_NOTIFICATION_V1, STATUS_NOTIFICATION_V2

Reason Code: 400

The specified ID is not consistent with the provided record, or the provided record is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 4-103 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 4-104 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 404

The provided record could not be found to update.

Reason Code: 409

A related record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Delete a Webhook — DELETE

This section describes the Delete a Webhook API.

Method

DELETE

URL

/api/delivery/v5/webhook/{id}

Allowed Scope(s)

webhook-rw

Request

This section describes the request parameters.

Table 4-105 Delete a Webhook— Request Parameters

Parameter	Required	Data Type	Description
id	yes	integer(\$int64) (path)	ID

Responses

This section describes the responses of the Delete a Webhook API.

Response Code: 204

Deletion succeeded.

Response Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Checks If a Webhook Exists — HEAD

This section describes the Checks if a Webhook Exists API.

Method

HEAD

URL

```
/api/delivery/v5/webhook/{id}
```

Allowed Scope(s)

```
webhook-rw
```

Request

This section describes the request parameters.

Table 4-106 Checks if a Webhook Exists — Request Parameters

Parameter	Required	Data Type	Description
id	yes	integer(\$int64) (path)	ID

Responses

This section describes the responses of the Checks if a Webhook Exists API.

Response Code: 200

The requested record exists.

Response Code: 404

The record could not be found.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Webhook Records By Attributes — GET

This section describes the Queries for Webhook Records API. This API returns a list of records matching the provided attributes.

Method

GET

URL

/api/delivery/v5/webhook/byattributes

Allowed Scope(s)

webhook-rw

Request

This section describes the request parameters.

Table 4-107 Queries for Webhook Records By Attributes — Request Parameters

Parameter	Required	Data Type	Description
id	NA	integer(\$int64) (path)	ID
credentialId	NA	string (query)	credential ID
description	NA	string (query)	description
url	NA	string (query)	URL
webhookId	NA	string (query)	webhook ID
webhookType	NA	string (query)	webhook type; Available values: STATUS_NOTIFICATION_V1, STATUS_NOTIFICATION_V2
orderBy	NA	string (query)	A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by " asc " (default) or " desc " to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.
limit	NA	integer (\$int32) (query)	Provides a maximum number of records to be returned.

Table 4-107 (Cont.) Queries for Webhook Records By Attributes — Request Parameters

Parameter	Required	Data Type	Description
offset	NA	integer (\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for Webhook Records By Attributes API.

Response Code: 200

The requested record exists.

The media type is application/json.

Example Value

```
[
  {
    "created": "2023-12-19T13:41:32.765Z",
    "createdBy": "string",
    "updated": "2023-12-19T13:41:32.765Z",
    "updatedBy": "string",
    "id": <id>,
    "credentialId": "string",
    "description": "string",
    "url": "string",
    "webhookId": "string",
    "webhookType": "STATUS_NOTIFICATION_V1"
  }
]
```

Schema — Webhook

The following table describes the elements of a webhook used to send delivery status notifications to an external service.

Table 4-108 Webhook — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.

Table 4-108 (Cont.) Webhook — Object

Element Name	Required	Data Type	Description
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
credentialId	Yes	string(32)	The credential ID.
description	NA	string(256)	The webhook description, if one has been defined.
url	Yes	string(1024)	The URL for the webhook.
webhookId	Yes	string(32)	The webhook ID. Webhook IDs must contain only letters, numbers, and underscores.
webhookType	Yes	string	Enum: STATUS_NOTIFICATION_V1, STATUS_NOTIFICATION_V2

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Data Retention Configuration Service

This section describes the Data Retention Configuration Service.

Add a New DataRetentionConfig — POST

This section describes the Add a New DataRetentionConfig API.

Method

POST

URL

/api/job/v5/dataretentionconfig

Allowed Scope(s)

job-rw

Request

This section describes the request parameters.

The request body is application/json.

The ModelField is null.

Table 4-109 Add a New DataRetentionConfig — Request Parameters

Parameter	Required	Data Type	Description
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is "child.line" would return the relationship named "child", and its child relationship named "line"), on the resulting model which should be included in the response or "all" if all associated relationships should be included.

Example Value

```
{
  "created": "2024-06-05T06:49:20.451Z",
  "createdBy": "string",
  "updated": "2024-06-05T06:49:20.451Z",
  "updatedBy": "string",
  "id": <id>,
  "dataType": "DELIVERY",
  "days": 99999999,
  "triggerHour": 23
}
```

Schema — DataRetention Configuration

The following table describes the elements of a configuration value used to determine retention for a piece of information.

Table 4-110 Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
dataType	Yes	string	The type of retained data. Enum: DELIVERY, QUOTE

Table 4-110 (Cont.) Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
days	NA	integer(\$int32)	The number of days to retain data of the assigned type.
triggerHour	Yes	integer(\$int32)	The hour each day, in UTC, when retained data beyond the retention interval should be removed.

Responses

This section describes the responses of the Add a New DataRetention Configuration API.

Reason Code: 200

The new record.

The media type is application/json.

Example Value

```
{
  "created": "2024-06-05T06:49:20.454Z",
  "createdBy": "string",
  "updated": "2024-06-05T06:49:20.454Z",
  "updatedBy": "string",
  "id": <id>,
  "dataType": "DELIVERY",
  "days": 99999999,
  "triggerHour": 23
}
```

Schema — DataRetention Configuration

The following table describes the elements of a configuration value used to determine retention for a piece of information.

Table 4-111 Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.

Table 4-111 (Cont.) Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
dataType	Yes	string	The type of retained data. Enum: DELIVERY, QUOTE
days	NA	integer(\$int32)	The number of days to retain data of the assigned type.
triggerHour	Yes	integer(\$int32)	The hour each day, in UTC, when retained data beyond the retention interval should be removed.

Reason Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 4-112 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known

Table 4-112 (Cont.) Mapper Error Data — Object

Element Name	Required	Data Type	Description
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 4-113 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 409

The record already exists or a related entity was not found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for DataRetention Configuration Records — GET

This section describes the Queries for DataRetention Configuration Records API. This API returns a list of records matching the provided criteria.

Method

GET

URL

`/api/job/v5/dataretentionconfig`

Allowed Scope(s)

`job-rw`

Request

This section describes the request parameters.

Table 4-114 Queries for DataRetention Configuration Records — Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, that is a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example {"child.name":"Jo"} would match Joe, John, and so on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is "@and1",</p>

Table 4-114 (Cont.) Queries for DataRetention Configuration Records — Request Parameters

Parameter	Required	Data Type	Description
orderBy	NA	string (query)	<p>"@and2", and so on) to ensure that they are uniquely named since any duplicates will be eliminated.</p> <p>A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by "asc" (default) or "desc" to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.</p>
expands	NA	string (query)	<p>A comma-delimited list of relationship paths, in dot-notated path form (that is "child.line" would return the relationship named "child", and its child relationship named "line"), on the resulting model which should be included in the response or "all" if all associated relationships should be included.</p>
limit	NA	integer(\$int32) (query)	Provides a maximum number of records to be returned.
offset	NA	integer(\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for DataRetention Configuration Records API.

Response Code: 200

The requested records.

The media type is application/json.

Example Value

```
[
  {
    "created": "2023-12-14T13:29:34.113Z",
    "createdBy": "string",
```

```
    "updated": "2023-12-14T13:29:34.113Z",
    "updatedBy": "string",
    "id": <id>,
    "credentialId": "string",
    "description": "string",
    "url": "string",
    "webhookId": "string",
    "webhookType": "STATUS_NOTIFICATION_V1"
  }
]
```

Schema — DataRetention Configuration

The following table describes the elements of a configuration value used to determine retention for a piece of information.

Table 4-115 Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
dataType	Yes	string	The type of retained data. Enum: DELIVERY, QUOTE
days	NA	integer(\$int32)	The number of days to retain data of the assigned type.
triggerHour	Yes	integer(\$int32)	The hour each day, in UTC, when retained data beyond the retention interval should be removed.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for DataRetention Configuration Records/Count — GET

This section describes the Queries for DataRetention Configuration Records/Count API. This API returns a count of records matching the provided criteria.

Method

GET

URL

`/api/job/v5/dataretentionconfig/count`

Allowed Scope(s)

`job-rw`

Request

This section describes the request parameters.

Table 4-116 Queries for DataRetention Configuration Records/Count — Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, i.e. a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example {"child.name":"Jo"} would match Joe, John, and so on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take</p>

Table 4-116 (Cont.) Queries for DataRetention Configuration Records/Count — Request Parameters

Parameter	Required	Data Type	Description
			care to suffix them with additional characters (that is "'@and1'", "'@and2'", and so on) to ensure that they are uniquely named since any duplicates will be eliminated.

Responses

This section describes the responses of the Queries for DataRetention Configuration Records/Count API.

Response Code: 200

The count of requested records.

The media type is application/json.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Get a DataRetention Configuration — GET

This section describes the Get a DataRetention Configuration API.

Method

GET

URL

```
/api/job/v5/dataretentionconfig/{id}
```

Allowed Scope(s)

job-rw

Request

This section describes the request parameters.

Table 4-117 Get a DataRetention Configuration — Request Parameters

Parameter	Required	Data Type	Description
id	yes	integer(\$int64) (path)	ID
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Responses

This section describes responses of the Get a DataRetention Configuration API.

Response Code: 200

The requested record.

The media type is applicaton/json.

Example Value

```
{
  "created": "2024-06-05T08:06:57.246Z",
  "createdBy": "string",
  "updated": "2024-06-05T08:06:57.246Z",
  "updatedBy": "string",
  "id": <id>,
  "dataType": "DELIVERY",
  "days": 99999999,
  "triggerHour": 23
}
```

Schema — DataRetention Configuration

The following table describes the elements of a configuration value used to determine retention for a piece of information.

Table 4-118 Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.

Table 4-118 (Cont.) Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
dataType	Yes	string	The type of retained data. Enum: DELIVERY, QUOTE
days	NA	integer(\$int32)	The number of days to retain data of the assigned type.
triggerHour	Yes	integer(\$int32)	The hour each day, in UTC, when retained data beyond the retention interval should be removed.

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Update a DataRetention Configuration — PUT

This section describes the Update a DataRetention Configuration API.

Method

PUT

URL

/api/job/v5/dataretentionconfig/{id}

Allowed Scope(s)

job-rw

Request

This section describes the request parameters.

The request body is application/json.

Table 4-119 Update a DataRetention Configuration — Request Parameters

Parameter	Required	Data Type	Description
id	yes	integer(\$int64) (path)	ID
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Example Value

```
{
  "created": "2024-06-05T08:50:00.771Z",
  "createdBy": "string",
  "updated": "2024-06-05T08:50:00.771Z",
  "updatedBy": "string",
  "id": <id>,
  "dataType": "DELIVERY",
  "days": 99999999,
  "triggerHour": 23
}
```

Schema — DataRetention Configuration

The following table describes the elements of a configuration value used to determine retention for a piece of information.

Table 4-120 Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID

Table 4-120 (Cont.) Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
dataType	Yes	string	The type of retained data. Enum: DELIVERY, QUOTE
days	NA	integer(\$int32)	The number of days to retain data of the assigned type.
triggerHour	Yes	integer(\$int32)	The hour each day, in UTC, when retained data beyond the retention interval should be removed.

Responses

This section describes the responses of the Update a DataRetention Configuration API.

Reason Code: 200

The updated record.

The media type application/json.

Example Value

```
{
  "created": "2024-06-05T08:50:00.773Z",
  "createdBy": "string",
  "updated": "2024-06-05T08:50:00.773Z",
  "updatedBy": "string",
  "id": <id>,
  "dataType": "DELIVERY",
  "days": 99999999,
  "triggerHour": 23
}
```

Schema — DataRetention Configuration

The following table describes the elements of a configuration value used to determine retention for a piece of information.

Table 4-121 Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.

Table 4-121 (Cont.) Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
dataType	Yes	string	The type of retained data. Enum: DELIVERY, QUOTE
days	NA	integer(\$int32)	The number of days to retain data of the assigned type.
triggerHour	Yes	integer(\$int32)	The hour each day, in UTC, when retained data beyond the retention interval should be removed.

Reason Code: 400

The specified ID is not consistent with the provided record, or the provided record is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 4-122 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 4-123 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 404

The provided record could not be found to update.

Reason Code: 409

A related record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Delete a DataRetention Configuration — DELETE

This section describes the Delete a DataRetention Configuration API.

Method

DELETE

URL

/api/job/v5/dataretentionconfig/{id}

Allowed Scope(s)

job-rw

Request

This section describes the request parameters.

Table 4-124 Delete a Webhook— Request Parameters

Parameter	Required	Data Type	Description
id	yes	integer(\$int64) (path)	ID

Responses

This section describes the responses of the Delete a DataRetention Configuration API.

Response Code: 204

Deletion succeeded.

Response Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Checks If a DataRetention Configuration Exists — HEAD

This section describes the Checks if a DataRetention Configuration Exists API.

Method

HEAD

URL

/api/job/v5/dataretentionconfig/{id}

Allowed Scope(s)

job-rw

Request

This section describes the request parameters.

Table 4-125 Checks if a DataRetention Configuration Exists — Request Parameters

Parameter	Required	Data Type	Description
id	yes	integer(\$int64) (path)	ID

Responses

This section describes the responses of the Checks if a DataRetention Configuration Exists API.

Response Code: 200

The requested record exists.

Response Code: 404

The record could not be found.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for DataRetention Configuration Records By Attributes — GET

This section describes the Queries for DataRetention Configuration Records API. This API returns a list of records matching the provided attributes.

Method

GET

URL

/api/job/v5/dataretentionconfig/byattributes

Allowed Scope(s)

job-rw

Request

This section describes the request parameters.

Table 4-126 Queries for DataRetention Configuration Records By Attributes — Request Parameters

Parameter	Required	Data Type	Description
id	NA	integer(\$int64) (query)	ID
datatype	NA	string (query)	Available values : DELIVERY, QUOTE
days	NA	integer(\$int32) (query)	days
triggerHour	NA	integer(\$int32) (query)	trigger hour
orderBy	NA	string (query)	A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by " asc " (default) or " desc " to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (i.e. " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or "all" if all associated relationships should be included.
limit	NA	integer(\$int32) (query)	Provides a maximum number of records to be returned.
offset	NA	integer(\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for DataRetention Configuration Records By Attributes API.

Response Code: 200

The requested record.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-06-05T09:43:29.656Z",
    "createdBy": "string",
    "updated": "2024-06-05T09:43:29.656Z",
    "updatedBy": "string",
    "id": <id>,
    "dataType": "DELIVERY",
    "days": 99999999,
    "triggerHour": 23
  }
]
```

Schema — DataRetention Configuration

The following table describes the elements of a configuration value used to determine retention for a piece of information.

Table 4-127 Schema — DataRetention Configuration

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
dataType	Yes	string	The type of retained data. Enum: DELIVERY, QUOTE
days	NA	integer(\$int32)	The number of days to retain data of the assigned type.
triggerHour	Yes	integer(\$int32)	The hour each day, in UTC, when retained data beyond the retention interval should be removed.

Response Code: 500

An error occurred during persistence or the request failed for some other reason.

Error Handling

When limits have been specified on fields, they are enforced on models and service models that are accepted by the application. When a limit is defined, it may be enforced via Java Bean Validation annotations or by the database itself.

Bean Validations

Limits enforced on fields, which violated, result in a response containing a list of "MapperErrorData" objects. These objects include:

- If defined, an error code
- An error message
- If appropriate, the value associated with the error
- If appropriate, the name of the field associated with the error
- The type of the error, which will be VALIDATION (an enumerated value) in this case

Error Representation

Errors related to validation conditions are always represented to the user as an HTTP 400 including an array of one or more MapperErrorData objects. MapperErrorData objects include the following fields:

Table 4-128 Fields — MapperErrorData Objects

Field	Description
column	Returns the JSON file column number the error occurred on, if known.
line	Returns the JSON file line number the error occurred on, if known.
errorCode	Returns the application-specific error code associated with the error, if known.
errorMessage	Returns the error message (not localized).
field	Returns an identifier of the associated field if known. This will be a path within a provided object, or some other field identifier.
value	Returns the JSON value that caused the error if known.
errorType	Return the error type. (one of INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION)
targetType	Returns the type the system expected the JSON node to be, if known.

Error Codes

A number of standard error codes are included in the services to deal with various conditions, their identifiers and descriptive message templates and include the following:

- 100 - Add entity was null.

- 101 - Add entity must not include an ID.
- 102 - <0> already exists.
- 103 - Related entity was not found.
- 104 - A constraint violation occurred on this, or a descendant, entity.
- 105 - Add request <0> ID must be greater than zero.
- 106 - <0> parent was not found.
- 107 - Child could not be added to <0>.
- 108 - <0> was not found.
- 109 - Bulk request was null.
- 110 - This endpoint accepts, at most, 1000 records.
- 111 - A limit must be specified when specifying an offset.
- 112 - Update entity was null.
- 113 - Update request ID does not match entity ID.
- 114 - Delete failed due to dependent data.
- 115 - Persist failed due to conflict on <0>.
- 116 - Update failed due to conflict on <0>.
- 117 - JSON parameter <0> does not match a field definition for entity.
- 118 - OrderBy parameter <0> does not match a field definition for entity.
- 119 - OrderBy order <0> is not valid.
- 120 - Unable to delete. At least one <0> is required.
- 199 - An error occurred during persistence.
- 200 - A JSON mapping error occurred.
- 201 - A JSON parsing error occurred.
- 202 - A validation error occurred.
- 203 - An invalid value was specified.
- 999 - Failed

Collect and Receive Error Codes

Additionally, Collect and Receive specific error codes have been defined. These codes may be returned to reflect transitive errors reaching carriers (transitive errors not detailed below utilize 999 from the list above):

- 1000 - Request could not be accepted as provided. (HTTP 400)
- 1002 - Request was not authorized. (HTTP 401)
- 1003 - A resource could not be found. (HTTP 404)
- 1004 - Too many requests have been submitted. (HTTP 429)
- 1005 - A server error occurred. (HTTP 500)
- 1006 - Service was unavailable. (HTTP 503)
- 1007 - Invalid or Missing Address

- 1008 - Invalid Pickup Time
- 1009 - Invalid Dropoff Time
- 1010 - Insufficient Couriers
- 1011 - Invalid Quote
- 1012 - Account Error
- 1013 - Request Timeout
- 1014 - Duplicate Delivery
- 1015 - Account not found
- 1016 - The entity was not in a status compatible with the requested action.
- 1017 - The delivery can not be canceled.
- 1018 - Delivery not found.

5

Foundation Service

This section describes the Foundation Service. The Foundation Service includes:

- [Location Service](#)
- [Package Dimension Service](#)

Note:

The Package Dimension Service can be used to add, replace, edit or delete the default values.

- [Reason Code Service](#)

Note:

The Reason Code Service can be used to add, replace, edit or delete the default values.

The Foundation Service contains the following OpenAPI endpoint, which can be used to obtain the specification for the services it provides.

`api/foundation/openapi`

Location Service

This section describes the Location Service.

Add a New Location — POST

This section describes the Add a New Location API. Use this service to add a new location record. This service is commonly used to set up Locations for the Collect and Receive User Interface to leverage when creating deliveries and viewing deliveries.

Method

POST

URL

`/api/foundation/v1/location`

Allowed Scope(s)

foundation-r
foundation-rw

Request

This section describes the request parameters.

The request body is application/json.

The ModelField is null.

Table 5-1 Queries for Webhook Records By Attributes — Request Parameters

Parameter	Required	Data Type	Description
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (i.e. " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Example Value

```
{
  "created": "2024-07-24T15:57:18.266Z",
  "createdBy": "string",
  "updated": "2024-07-24T15:57:18.266Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "deliveryInstructions": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "latitude": 999999999,
  "locTypeDescription": "string",
  "locationCode": "string",
  "locationDesc": "string",
  "longitude": 999999999,
```

```

    "operatingHours": "string",
    "phoneNumber": "string",
    "postalCode": "string",
    "state": "string",
    "timezone": "string"
  }
  "string"
}

```

Schema — Location

This section describes the Location schema.

Table 5-2 Location — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
address1	Yes	string(254)	The address line 1.
address2	NA	string(254)	The address line 2.
address3	NA	string(254)	The address line 3.
address4	NA	string(254)	The address line 4.
addressType	NA	string	The address type identifier. Enum: RESIDENTIAL, BUSINESS
apartment	NA	string(254)	The apartment at the location.
city	Yes	string(254)	The city of the location.
country	Yes	string(254)	The ISO-3166 country code.
deliveryInstructions	NA	string(254)	Instructions for courier providers when picking up/delivering.
emailAddress	NA	string(254)	The email address.
firstName	NA	string(254)	The first (i.e. given) name.
lastName	NA	string(254)	The last (i.e. family) name.
latitude	NA	number	The latitude of the location.
locTypeDescription	NA	string(40)	The type of location (i.e. store, warehouse, office, etc.).
locationCode	Yes	string(10)	The unique code for the location.
locationDesc	Yes	string(254)	The location description.
longitude	NA	number	The longitude of the location.
operatingHours	NA	string(254)	The operating hours of the location.

Table 5-2 (Cont.) Location — Object

Element Name	Required	Data Type	Description
phoneNumber	NA	string(50)	The phone contact number.
postalCode	NA	string(20)	The postal code of the location.
state	NA	string(254)	The ISO 3166 state, province, or territory code.
timezone	Yes	string(40)	Identifies the timezone by continent/region and area of the user.

Responses

This section describes the responses of the Add a New Location API.

Response Code: 200

The new record.

The media type is application/json.

Example Value

```
{
  "created": "2024-07-24T15:57:18.272Z",
  "createdBy": "string",
  "updated": "2024-07-24T15:57:18.272Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "deliveryInstructions": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "latitude": 999999999,
  "locTypeDescription": "string",
  "locationCode": "string",
  "locationDesc": "string",
  "longitude": 999999999,
  "operatingHours": "string",
  "phoneNumber": "string",
  "postalCode": "string",
  "state": "string",
  "timezone": "string"
}
```

Schema — Location

This section describes the Location schema.

Table 5-3 Location — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
address1	Yes	string(254)	The address line 1.
address2	NA	string(254)	The address line 2.
address3	NA	string(254)	The address line 3.
address4	NA	string(254)	The address line 4.
addressType	NA	string	The address type identifier. Enum: RESIDENTIAL, BUSINESS
apartment	NA	string(254)	The apartment at the location.
city	Yes	string(254)	The city of the location.
country	Yes	string(254)	The ISO-3166 country code.
deliveryInstructions	NA	string(254)	Instructions for courier providers when picking up/delivering.
emailAddress	NA	string(254)	The email address.
firstName	NA	string(254)	The first (i.e. given) name.
lastName	NA	string(254)	The last (i.e. family) name.
latitude	NA	number	The latitude of the location.
locTypeDescription	NA	string(40)	The type of location (i.e. store, warehouse, office, etc.).
locationCode	Yes	string(10)	The unique code for the location.
locationDesc	Yes	string(254)	The location description.
longitude	NA	number	The longitude of the location.
operatingHours	NA	string(254)	The operating hours of the location.
phoneNumber	NA	string(50)	The phone contact number.
postalCode	NA	string(20)	The postal code of the location.
state	NA	string(254)	The ISO 3166 state, province, or territory code.
timezone	Yes	string(40)	Identifies the timezone by continent/region and area of the user.

Reason Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 5-4 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 5-5 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 409

The record already exists or a related entity was not found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Location Records — GET

This section describes the Queries for Location Records API. This service returns a list of location records and associated location details matching the provided criteria.

Method

GET

URL

/api/foundation/v1/location

Allowed Scope(s)

foundation-r

Request

Table 5-6 Queries for Location Records— Request Parameters

where	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, i.e. a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example, {"child.name":"Jo"} would match Joe, John, and so on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":"100.00"}.</p> <p>If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example, {"id":["1", "2", "3"]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, e.g. {"id":{"from":"0","to":"100"}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (i.e. as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (i.e. "@and1", "@and2", and so on) to ensure that they are uniquely named since any duplicates will be eliminated.</p>
orderBy	string (query)	<p>A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by "asc" (default) or "desc" to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.</p>

Table 5-6 (Cont.) Queries for Location Records— Request Parameters

expand	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.
limit	integer(\$int32) (query)	Provides a maximum number of records to be returned.
offset	integer(\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for Quote Records Count endpoint.

Response Code: 200

The requested records.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-07-24T15:59:42.587Z",
    "createdBy": "string",
    "updated": "2024-07-24T15:59:42.587Z",
    "updatedBy": "string",
    "id": <id>,
    "address1": "string",
    "address2": "string",
    "address3": "string",
    "address4": "string",
    "addressType": "RESIDENTIAL",
    "apartment": "string",
    "city": "string",
    "country": "string",
    "deliveryInstructions": "string",
    "emailAddress": "string",
    "firstName": "string",
    "lastName": "string",
    "latitude": 999999999,
    "locTypeDescription": "string",
    "locationCode": "string",
    "locationDesc": "string",
    "longitude": 999999999,
    "operatingHours": "string",
    "phoneNumber": "string",
```

```

        "postalCode": "string",
        "state": "string",
        "timezone": "string"
    }
]

```

Schema — Location

This section describes the Location schema.

Table 5-7 Location — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
address1	Yes	string(254)	The address line 1.
address2	NA	string(254)	The address line 2.
address3	NA	string(254)	The address line 3.
address4	NA	string(254)	The address line 4.
addressType	NA	string	The address type identifier. Enum: RESIDENTIAL, BUSINESS
apartment	NA	string(254)	The apartment at the location.
city	Yes	string(254)	The city of the location.
country	Yes	string(254)	The ISO-3166 country code.
deliveryInstructions	NA	string(254)	Instructions for courier providers when picking up/delivering.
emailAddress	NA	string(254)	The email address.
firstName	NA	string(254)	The first (i.e. given) name.
lastName	NA	string(254)	The last (i.e. family) name.
latitude	NA	number	The latitude of the location.
locTypeDescription	NA	string(40)	The type of location (i.e. store, warehouse, office, etc.).
locationCode	Yes	string(10)	The unique code for the location.
locationDesc	Yes	string(254)	The location description.
longitude	NA	number	The longitude of the location.
operatingHours	NA	string(254)	The operating hours of the location.
phoneNumber	NA	string(50)	The phone contact number.
postalCode	NA	string(20)	The postal code of the location.
state	NA	string(254)	The ISO 3166 state, province, or territory code.

Table 5-7 (Cont.) Location — Object

Element Name	Required	Data Type	Description
timezone	Yes	string(40)	Identifies the timezone by continent/region and area of the user.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Performs a Bulk Action Using a Collection of Locations — POST

This section describes the Performs a Bulk Action Using a Collection of Locations API. This service performs a bulk action (Add, Add or Update, Delete, Update) using a Collection of Locations services.

Method

POST

URL

/api/foundation/v1/location/bulk

Allowed Scope(s)

foundation-r
foundation-rw

Request

The request body is application/json.

Requests a lists of records to be loaded.

Table 5-8 Performs a Bulk Action Using a Collection of Locations — Request Parameters

Parameter	Required	Data Type	Description
action	NA	string (query)	Provides an action to be used performing the bulk operation. If not specified, the default of ADD_OR_UPDATE is used. <i>Available values</i> : ADD, ADD_OR_UPDATE, DELETE, UPDATE

Example Value

```
[
  {
    "created": "2024-07-24T16:00:54.793Z",
    "createdBy": "string",
    "updated": "2024-07-24T16:00:54.793Z",
    "updatedBy": "string",
    "id": <id>,
    "address1": "string",
    "address2": "string",
    "address3": "string",
    "address4": "string",
    "addressType": "RESIDENTIAL",
    "apartment": "string",
    "city": "string",
    "country": "string",
    "deliveryInstructions": "string",
    "emailAddress": "string",
    "firstName": "string",
    "lastName": "string",
    "latitude": 999999999,
    "locTypeDescription": "string",
    "locationCode": "string",
    "locationDesc": "string",
    "longitude": 999999999,
    "operatingHours": "string",
    "phoneNumber": "string",
    "postalCode": "string",
    "state": "string",
    "timezone": "string"
  }
]
```

Schema — Location

This section describes the Location schema.

Table 5-9 Location — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id

Table 5-9 (Cont.) Location — Object

Element Name	Required	Data Type	Description
address1	Yes	string(254)	The address line 1.
address2	NA	string(254)	The address line 2.
address3	NA	string(254)	The address line 3.
address4	NA	string(254)	The address line 4.
addressType	NA	string	The address type identifier. Enum: RESIDENTIAL, BUSINESS
apartment	NA	string(254)	The apartment at the location.
city	Yes	string(254)	The city of the location.
country	Yes	string(254)	The ISO-3166 country code.
deliveryInstructions	NA	string(254)	Instructions for courier providers when picking up/delivering.
emailAddress	NA	string(254)	The email address.
firstName	NA	string(254)	The first (i.e. given) name.
lastName	NA	string(254)	The last (i.e. family) name.
latitude	NA	number	The latitude of the location.
locTypeDescription	NA	string(40)	The type of location (i.e. store, warehouse, office, etc.).
locationCode	Yes	string(10)	The unique code for the location.
locationDesc	Yes	string(254)	The location description.
longitude	NA	number	The longitude of the location.
operatingHours	NA	string(254)	The operating hours of the location.
phoneNumber	NA	string(50)	The phone contact number.
postalCode	NA	string(20)	The postal code of the location.
state	NA	string(254)	The ISO 3166 state, province, or territory code.
timezone	Yes	string(40)	Identifies the timezone by continent/region and area of the user.

Responses

This section describes the responses of the Performs a Bulk Action Using a Collection of Locations endpoint.

Response Code: 200

The result of the loading operation.

The media type is application/json.

Example Value

```
{
  "errorRecords": [
    {
      "index": 0,
      "error": "CONFLICT"
    }
  ],
  "successCount": 0
}
```

Schema — Bulk Operation Result

The following table describes the elements of a Bulk Operation, describing the result of a bulk operation.

Table 5-10 Bulk Operation Result — Object

Element Name	Required	Data Type	Description
errorRecords	NA	object	A list of ErrorRecord objects describing any records which were not successfully processed.
successCount	NA	integer(\$int32)	The count of the records which were successfully processed.

The following table describes an object describing an error which occurred during a bulk operation.

Table 5-11 Error Record— Object

Element Name	Required	Data Type	Description
index	NA	integer(\$int32)	The 0-based index of the associated record which was not processed.

Table 5-11 (Cont.) Error Record— Object

Element Name	Required	Data Type	Description
error	NA	string	<p>An enumeration describing the result of a bulk action which was not successfully completed.</p> <ul style="list-style-type: none">• CONFLICT - The record was provided with criteria which conflicts with another record.• DOES_NOT_EXIST - The record was provided with an action which required it to exist, but the record was not found.• ALREADY_EXISTS - The record was provided with an action which required it not to exist, but the record was found.• RELATED_DOES_NOT_EXIST - The record was provided without a required related relationship.• INVALID - The record was invalid. Cases for this include:<ul style="list-style-type: none">– The record was provided as an ADD but included an ID.– The record was provided as a DELETE or UPDATE but lacked an ID.– The record was provided with an illegal/invalid field value.• UNKNOWN - Some other unexpected exception occurred. <p>Enum: [CONFLICT, DOES_NOT_EXIST, ALREADY_EXISTS, RELATED_DOES_NOT_EXIST, INVALID, UNKNOWN]</p>

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
```

```

        "description": "string"
      },
      "field": "string",
      "value": "string",
      "errorType": "INTEGRATION"
    }
  ]

```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 5-12 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 5-13 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 413

Too many records were provided.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Location Records/Count — GET

This section describes the Queries for Location Records/Count API. This API returns a count of records matching the provided criteria.

Method

GET

URL

`/api/foundation/v1/location/count`

Allowed Scope(s)

`foundation-r`

Request

Table 5-14 Queries for Location Records/Count — Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, i.e. a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example, {"child.name":"Jo"} would match Joe, John, etc. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example, {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example, {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (i.e. as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (i.e. "@and1", "@and2", etc.) to ensure that they are uniquely named since any duplicates will be eliminated</p>

Responses

This section describes the responses of the Queries for Location Records Count endpoint.

Response Code: 200

The count of the requested records.

The media type is application/json.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Get a Location — GET

This section describes the Get a Location API. Use this service to retrieve a specific location record and the associated details.

Method

GET

URL

/api/foundation/v1/location/{id}

Allowed Scope(s)

foundation-r

Request

This section describes the request parameters.

Table 5-15 Get a Location — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64)	ID
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Responses

This section describes the responses of the Get a Location API.

Response Code: 200

- The requested record.
- The media type is application/json.

Example Value

```
{
  "created": "2024-07-24T16:02:05.369Z",
  "createdBy": "string",
  "updated": "2024-07-24T16:02:05.369Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "deliveryInstructions": "string",
  "emailAddress": "string",
  "firstName": "string",
  "lastName": "string",
  "latitude": 999999999,
  "locTypeDescription": "string",
  "locationCode": "string",
  "locationDesc": "string",
  "longitude": 999999999,
  "operatingHours": "string",
  "phoneNumber": "string",
  "postalCode": "string",
  "state": "string",
  "timezone": "string"
}
```

Schema — Location

This section describes the Location schema.

Table 5-16 Location — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.

Table 5-16 (Cont.) Location — Object

Element Name	Required	Data Type	Description
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
address1	Yes	string(254)	The address line 1.
address2	NA	string(254)	The address line 2.
address3	NA	string(254)	The address line 3.
address4	NA	string(254)	The address line 4.
addressType	NA	string	The address type identifier. Enum: RESIDENTIAL, BUSINESS
apartment	NA	string(254)	The apartment at the location.
city	Yes	string(254)	The city of the location.
country	Yes	string(254)	The ISO-3166 country code.
deliveryInstructions	NA	string(254)	Instructions for courier providers when picking up/delivering.
emailAddress	NA	string(254)	The email address.
firstName	NA	string(254)	The first (i.e. given) name.
lastName	NA	string(254)	The last (i.e. family) name.
latitude	NA	number	The latitude of the location.
locTypeDescription	NA	string(40)	The type of location (i.e. store, warehouse, office, etc.).
locationCode	Yes	string(10)	The unique code for the location.
locationDesc	Yes	string(254)	The location description.
longitude	NA	number	The longitude of the location.
operatingHours	NA	string(254)	The operating hours of the location.
phoneNumber	NA	string(50)	The phone contact number.
postalCode	NA	string(20)	The postal code of the location.
state	NA	string(254)	The ISO 3166 state, province, or territory code.
timezone	Yes	string(40)	Identifies the timezone by continent/region and area of the user.

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Update a Location — PUT

This section describes the Update a Location API. Use this service to update a specific location's attributes.

Method

PUT

URL

/api/foundation/v1/location/{id}

Allowed Scope(s)

foundation-r
foundation-rw

Request

This section describes the request parameters.
The request body is application/json.

Table 5-17 Update a Location — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is," child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Example Value

```
{
  "created": "2024-07-24T16:02:58.049Z",
  "createdBy": "string",
  "updated": "2024-07-24T16:02:58.049Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
```

```

    "address2": "string",
    "address3": "string",
    "address4": "string",
    "addressType": "RESIDENTIAL",
    "apartment": "string",
    "city": "string",
    "country": "string",
    "deliveryInstructions": "string",
    "emailAddress": "string",
    "firstName": "string",
    "lastName": "string",
    "latitude": 999999999,
    "locTypeDescription": "string",
    "locationCode": "string",
    "locationDesc": "string",
    "longitude": 999999999,
    "operatingHours": "string",
    "phoneNumber": "string",
    "postalCode": "string",
    "state": "string",
    "timezone": "string"
  }

```

Schema — Location

This section describes the Location schema.

Table 5-18 Location — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
address1	Yes	string(254)	The address line 1.
address2	NA	string(254)	The address line 2.
address3	NA	string(254)	The address line 3.
address4	NA	string(254)	The address line 4.
addressType	NA	string	The address type identifier. Enum: RESIDENTIAL, BUSINESS
apartment	NA	string(254)	The apartment at the location.
city	Yes	string(254)	The city of the location.
country	Yes	string(254)	The ISO-3166 country code.
deliveryInstructions	NA	string(254)	Instructions for courier providers when picking up/delivering.

Table 5-18 (Cont.) Location — Object

Element Name	Required	Data Type	Description
emailAddress	NA	string(254)	The email address.
firstName	NA	string(254)	The first (i.e. given) name.
lastName	NA	string(254)	The last (i.e. family) name.
latitude	NA	number	The latitude of the location.
locTypeDescription	NA	string(40)	The type of location (i.e. store, warehouse, office, etc.).
locationCode	Yes	string(10)	The unique code for the location.
locationDesc	Yes	string(254)	The location description.
longitude	NA	number	The longitude of the location.
operatingHours	NA	string(254)	The operating hours of the location.
phoneNumber	NA	string(50)	The phone contact number.
postalCode	NA	string(20)	The postal code of the location.
state	NA	string(254)	The ISO 3166 state, province, or territory code.
timezone	Yes	string(40)	Identifies the timezone by continent/region and area of the user.

Responses

This section describes the responses of the Update a Location API.

Response Code: 200

The updated record.

The media type is application/json.

Example Value

```
{
  "created": "2024-07-24T16:02:58.056Z",
  "createdBy": "string",
  "updated": "2024-07-24T16:02:58.056Z",
  "updatedBy": "string",
  "id": <id>,
  "address1": "string",
  "address2": "string",
  "address3": "string",
  "address4": "string",
  "addressType": "RESIDENTIAL",
  "apartment": "string",
  "city": "string",
  "country": "string",
  "deliveryInstructions": "string",
  "emailAddress": "string",
```



```

    "firstName": "string",
    "lastName": "string",
    "latitude": 999999999,
    "locTypeDescription": "string",
    "locationCode": "string",
    "locationDesc": "string",
    "longitude": 999999999,
    "operatingHours": "string",
    "phoneNumber": "string",
    "postalCode": "string",
    "state": "string",
    "timezone": "string"
  }

```

Schema — Location

This section describes the Location schema.

Table 5-19 Location — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
address1	Yes	string(254)	The address line 1.
address2	NA	string(254)	The address line 2.
address3	NA	string(254)	The address line 3.
address4	NA	string(254)	The address line 4.
addressType	NA	string	The address type identifier. Enum: RESIDENTIAL, BUSINESS
apartment	NA	string(254)	The apartment at the location.
city	Yes	string(254)	The city of the location.
country	Yes	string(254)	The ISO-3166 country code.
deliveryInstructions	NA	string(254)	Instructions for courier providers when picking up/delivering.
emailAddress	NA	string(254)	The email address.
firstName	NA	string(254)	The first (i.e. given) name.
lastName	NA	string(254)	The last (i.e. family) name.
latitude	NA	number	The latitude of the location.
locTypeDescription	NA	string(40)	The type of location (i.e. store, warehouse, office, etc.).
locationCode	Yes	string(10)	The unique code for the location.

Table 5-19 (Cont.) Location — Object

Element Name	Required	Data Type	Description
locationDesc	Yes	string(254)	The location description.
longitude	NA	number	The longitude of the location.
operatingHours	NA	string(254)	The operating hours of the location.
phoneNumber	NA	string(50)	The phone contact number.
postalCode	NA	string(20)	The postal code of the location.
state	NA	string(254)	The ISO 3166 state, province, or territory code.
timezone	Yes	string(40)	Identifies the timezone by continent/region and area of the user.

Reason Code: 400

The specified ID is not consistent with the provided record, or the provided record is invalid

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 5-20 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known

Table 5-20 (Cont.) Mapper Error Data — Object

Element Name	Required	Data Type	Description
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 5-21 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 404

The provided record could not be found to update.

Reason Code: 409

A related record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Delete a Location — DELETE

This section describes the Delete a Location API. Use this service to delete a location record and the associated details.

Method

DELETE

URL

```
/api/foundation/v1/location/{id}
```

Allowed Scope(s)

```
foundation-r  
foundation-rw
```

Request

This section describes the request parameters.

Table 5-22 Delete a Location — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID

Responses

This section describes the responses of the Delete a Location API.

Reason Code: 204

Deletion succeeded.

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Checks if a Location Exists — HEAD

This section describes the Checks if a Location Exists API.

Method

HEAD

URL

```
/api/foundation/v1/location/{id}
```

Allowed Scope(s)

```
foundation-r
```

Request

This section describes the request parameters.

Table 5-23 Checks if a Location Exists — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID

Responses

This section describes the responses of the Checks if a Location Exists API.

Reason Code: 200

The requested record exists.

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Location Records by Attributes— GET

This section describes the Queries for Location Records by Attributes API. This API returns a list of records matching the provided criteria.

Method

GET

URL

/api/foundation/v1/location/byattributes

Allowed Scope(s)

foundation-r

Request

Table 5-24 Queries for Location Records by Attributes — Request Parameters

Parameter	Required	Data Type	Description
id	NA	integer(\$int64) (query)	id
address1	NA	string (query)	address1
address2	NA	string (query)	address2
address3	NA	string (query)	address3
address4	NA	string (query)	address4
addressType	NA	string (query)	Available values : RESIDENTIAL, BUSINESS
city	NA	string (query)	city
country	NA	string (query)	country
deliveryInstructions	NA	string (query)	deliveryInstructions
email	NA	string (query)	email
firstName	NA	string (query)	firstName
lastName	NA	string (query)	lastName
latitude	NA	number (query)	latitude
locTypeDescription	NA	string (query)	locTypeDescription
locationCode	NA	string (query)	locationCode
locationDesc	NA	string (query)	locationDesc
longitude	NA	number (query)	longitude
operatingHours	NA	string (query)	operatingHours
phoneNumber	NA	string (query)	phoneNumber
postalCode	NA	string (query)	postalCode
state	NA	string (query)	state
suite	NA	string (query)	suite
timezone	NA	string (query)	timezone

Table 5-24 (Cont.) Queries for Location Records by Attributes — Request Parameters

Parameter	Required	Data Type	Description
orderBy	NA	string (query)	A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by " asc " (default) or " desc " to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.
limit	NA	integer(\$int32) (query)	Provides a maximum number of records to be returned.
offset	NA	integer(\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for Location Records by Attributes endpoint.

Response Code: 200

The requested records.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-07-24T16:05:15.787Z",
    "createdBy": "string",
    "updated": "2024-07-24T16:05:15.787Z",
    "updatedBy": "string",
    "id": <id>,
    "address1": "string",
    "address2": "string",
    "address3": "string",
```

```

    "address4": "string",
    "addressType": "RESIDENTIAL",
    "apartment": "string",
    "city": "string",
    "country": "string",
    "deliveryInstructions": "string",
    "emailAddress": "string",
    "firstName": "string",
    "lastName": "string",
    "latitude": 999999999,
    "locTypeDescription": "string",
    "locationCode": "string",
    "locationDesc": "string",
    "longitude": 999999999,
    "operatingHours": "string",
    "phoneNumber": "string",
    "postalCode": "string",
    "state": "string",
    "timezone": "string"
  }
]

```

Schema — Location

This section describes the Location schema.

Table 5-25 Location — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
address1	Yes	string(254)	The address line 1.
address2	NA	string(254)	The address line 2.
address3	NA	string(254)	The address line 3.
address4	NA	string(254)	The address line 4.
addressType	NA	string	The address type identifier. Enum: RESIDENTIAL, BUSINESS
apartment	NA	string(254)	The apartment at the location.
city	Yes	string(254)	The city of the location.
country	Yes	string(254)	The ISO-3166 country code.
deliveryInstructions	NA	string(254)	Instructions for courier providers when picking up/delivering.
emailAddress	NA	string(254)	The email address.

Table 5-25 (Cont.) Location — Object

Element Name	Required	Data Type	Description
firstName	NA	string(254)	The first (i.e. given) name.
lastName	NA	string(254)	The last (i.e. family) name.
latitude	NA	number	The latitude of the location.
locTypeDescription	NA	string(40)	The type of location (i.e. store, warehouse, office, etc.).
locationCode	Yes	string(10)	The unique code for the location.
locationDesc	Yes	string(254)	The location description.
longitude	NA	number	The longitude of the location.
operatingHours	NA	string(254)	The operating hours of the location.
phoneNumber	NA	string(50)	The phone contact number.
postalCode	NA	string(20)	The postal code of the location.
state	NA	string(254)	The ISO 3166 state, province, or territory code.
timezone	Yes	string(40)	Identifies the timezone by continent/region and area of the user.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Package Dimension Service

This section describes the Package Dimension Service.

**Note:**

The Package Dimension Service can be used to add, replace, edit or delete the default values.

Add a New PackageDimension — POST

This section describes the Add a New PackageDimension API. This service creates a package dimension code which associates a set of dimensions to a package description that can be configured by locale. This is also leveraged by the Collect and Receive user interface to allow users to select a package size description when creating a delivery, instead of entering specific dimensions.

Method

POST

URL

/api/foundation/v1/package/dimension

Allowed Scope(s)

foundation-r
foundation-rw

Request

This section describes the request parameters.

The request body is application/json.

The ModelField is null.

Table 5-26 Add a New PackageDimension — Request Parameters

Parameter	Required	Data Type	Description
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is," child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Responses

This section describes the responses of the Add a PackageDimension API.

Response Code: 200

The new record.

The media type is application/json.

Example Value

```
{
  "created": "2024-05-29T13:29:25.265Z",
  "createdBy": "string",
  "updated": "2024-05-29T13:29:25.265Z",
  "updatedBy": "string",
  "id": <id>,

```

```

    "descriptions": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "height": 9999,
    "length": 9999,
    "packDimensionCode": "string",
    "weight": 1000000000000000000,
    "width": 9999
  }
}

```

Schema — Package Dimension

This section describes the Package Dimension schema.

Table 5-27 Package Dimension — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
height	Yes	integer(\$int32)	The height of the package in centimeters.
length	Yes	integer(\$int32)	The length of the package in centimeters.
packDimensionCode	Yes	string	The package dimension code.
weight	Yes	number	The weight of the package in kilograms.
width	Yes	integer(\$int32)	The width of the package in centimeters.

Table 5-28 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions. For example, <pre>{ "en-US": "small", "de-DE": "klein", "fr-FR": "petit" }</pre>

Reason Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[  
  {  
    "column": 0,  
    "line": 0,  
    "errorCode": 0,  
    "errorMessage": "string",  
    "moreInfo": {  
      "code": "string",  
      "description": "string"  
    },  
    "field": "string",  
    "value": "string",  
    "errorType": "INTEGRATION"  
  }  
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 5-29 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known

Table 5-29 (Cont.) Mapper Error Data — Object

Element Name	Required	Data Type	Description
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 5-30 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 409

The record already exists or a related entity was not found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for PackageDimension Records— GET

This section describes the Queries for PackageDimension Records API. This service returns a list of package dimension records with associated package dimension details and descriptions matching the provided criteria.

Method

GET

URL

`/api/foundation/v1/package/dimension`

Allowed Scope(s)

`foundation-r`

Request

This section describes the request parameters.

Table 5-31 Queries for PackageDimension Records — Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, i.e. a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example, {"child.name":"Jo"} would match Joe, John, etc. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example, {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example, {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is, as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is, "@and1", "@and2")</p>

Table 5-31 (Cont.) Queries for PackageDimension Records — Request Parameters

Parameter	Required	Data Type	Description
orderBy	NA	string (query)	A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by "asc" (default) or "desc" to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, "child.line" would return the relationship named "child", and its child relationship named "line"), on the resulting model which should be included in the response or "all" if all associated relationships should be included.
limit	NA	integer(\$int32) (query)	Provides a maximum number of records to be returned.
offset	NA	integer(\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for PackageDimension Records API.

Response Code: 200

The requested record.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-05-28T09:14:28.643Z",
    "createdBy": "string",
    "updated": "2024-05-28T09:14:28.643Z",
    "updatedBy": "string",
    "id": <id>,
    "descriptions": {
      "additionalProp1": "string",
      "additionalProp2": "string",
```



```

        "additionalProp3": "string"
      },
      "height": 9999,
      "length": 9999,
      "packDimensionCode": "string",
      "weight": 10000000000000000,
      "width": 9999
    }
  ]

```

Schema — Package Dimension

This section describes the Package Dimension schema.

Table 5-32 Package Dimension — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
height	Yes	integer(\$int32)	The height of the package in centimeters.
length	Yes	integer(\$int32)	The length of the package in centimeters.
packDimensionCode	Yes	string	The package dimension code.
weight	Yes	number	The weight of the package in kilograms.
width	Yes	integer(\$int32)	The width of the package in centimeters.

Table 5-33 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions. For example, <pre>{ "en-US": "small", "de-DE": "klein", "fr-FR": "petit" }</pre>

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Performs a Bulk Action Using a Collection of PackageDimension — GET

This section describes the Performs a Bulk Action Using a Collection of PackageDimension API. This service performs a bulk action (Add, Add or Update, Delete, Update) using a collection of PackageDimension services.

Method

POST

URL

/api/foundation/v1/package/dimension/bulk

Allowed Scope(s)

foundation-r

Request

This section describes the request parameters.

The request body is application/json.

A list of records to be loaded.

Table 5-34 Performs a Bulk Action Using a Collection of PackageDimension — Request Parameters

Parameter	Required	Data Type	Description
action	NA	string (query)	Provides an action to be used performing the bulk operation. If not specified, the default of ADD_OR_UPDATE is used. Available values : ADD, ADD_OR_UPDATE, DELETE, UPDATE

Example Value

```
[
  {
    "created": "2024-05-28T10:26:04.369Z",
    "createdBy": "string",
    "updated": "2024-05-28T10:26:04.369Z",
    "updatedBy": "string",
    "id": <id>,
    "descriptions": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "height": 9999,
    "length": 9999,
    "packDimensionCode": "string",
    "weight": 10000000000000000,
    "width": 9999
  }
]
```

Schema — Package Dimension

This section describes the Package Dimension schema.

Table 5-35 Package Dimension — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.

Table 5-35 (Cont.) Package Dimension — Object

Element Name	Required	Data Type	Description
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
height	Yes	integer(\$int32)	The height of the package in centimeters.
length	Yes	integer(\$int32)	The length of the package in centimeters.
packDimensionCode	Yes	string	The package dimension code.
weight	Yes	number	The weight of the package in kilograms.
width	Yes	integer(\$int32)	The width of the package in centimeters.

Table 5-36 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions. For example, <pre>{ "en-US": "small", "de-DE": "klein", "fr-FR": "petit" }</pre>

Responses

This section describes the responses of the Performs a Bulk Action using a Collection of PackageDimension API.

Response Code: 200

The result of the loading operation.

The media type is application/json.

Example Value

```
{
  "errorRecords": [
    {
      "index": 0,
      "error": "CONFLICT"
    }
  ],
  "successCount": 0
}
```

Schema — Bulk Operation Result

The following table describes the elements of a Bulk Operation, describing the result of a bulk operation.

Table 5-37 Bulk Operation Result — Object

Element Name	Required	Data Type	Description
errorRecords	NA	object	A list of ErrorRecord objects describing any records which were not successfully processed.
successCount	NA	integer(\$int32)	The count of the records which were successfully processed.

The following table describes an object describing an error which occurred during a bulk operation.

Table 5-38 Error Record— Object

Element Name	Required	Data Type	Description
index	NA	integer(\$int32)	The 0-based index of the associated record which was not processed.

Table 5-38 (Cont.) Error Record— Object

Element Name	Required	Data Type	Description
error	NA	string	<p>An enumeration describing the result of a bulk action which was not successfully completed.</p> <ul style="list-style-type: none">• CONFLICT - The record was provided with criteria which conflicts with another record.• DOES_NOT_EXIST - The record was provided with an action which required it to exist, but the record was not found.• ALREADY_EXISTS - The record was provided with an action which required it not to exist, but the record was found.• RELATED_DOES_NOT_EXIST - The record was provided without a required related relationship.• INVALID - The record was invalid. Cases for this include:<ul style="list-style-type: none">– The record was provided as an ADD but included an ID.– The record was provided as a DELETE or UPDATE but lacked an ID.– The record was provided with an illegal/invalid field value.• UNKNOWN - Some other unexpected exception occurred. <p>Enum: [CONFLICT, DOES_NOT_EXIST, ALREADY_EXISTS, RELATED_DOES_NOT_EXIST, INVALID, UNKNOWN]</p>

Reason Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
```

```
        "description": "string"
      },
      "field": "string",
      "value": "string",
      "errorType": "INTEGRATION"
    }
  ]
}
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 5-39 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 5-40 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 413

Too many records were provided.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for PackageDimension Records/Count— GET

This section describes the Queries for PackageDimension Records/Count API. This API returns a count of records matching the provided criteria.

Method

GET

URL

`/api/foundation/v1/package/dimension/count`

Allowed Scope(s)

`foundation-r`

Request

This section describes the request parameters.

Table 5-41 Queries for PackageDimension Records/Count — Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, that is, a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example, {"child.name":"Jo"} would match Joe, John, and so on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example, {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example, {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is, as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is, "@and1", "@and2", and so on) to ensure that they are uniquely named since any duplicates will be eliminated.</p>

Responses

This section describes the responses of the Queries for PackageDimensions Records/Count API.

Response Code: 200

The count of requested records.

The media type is application/json.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Get a PackageDimension — GET

This section describes the Get a PackageDimension API. Use this service to retrieve a specific package dimension record and the associated dimensions and descriptions.

Method

GET

URL

/api/foundation/v1/package/dimension/{id}

Allowed Scope(s)

foundation-r

Request

This section describes the request parameters.

Table 5-42 Get a PackageDimension — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Responses

This section describes the responses of the Get a PackageDimension API.

Response Code: 200

- The requested record.
- The media type is application/json.

Example Value

```
{
  "created": "2024-05-29T13:29:25.265Z",
  "createdBy": "string",
  "updated": "2024-05-29T13:29:25.265Z",
  "updatedBy": "string",
  "id": <id>,
  "descriptions": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "height": 9999,
  "length": 9999,
  "packDimensionCode": "string",
  "weight": 10000000000000000,
  "width": 9999
}
```

Schema — Package Dimension

This section describes the Package Dimension schema.

Table 5-43 Package Dimension — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
height	Yes	integer(\$int32)	The height of the package in centimeters.
length	Yes	integer(\$int32)	The length of the package in centimeters.
packDimensionCode	Yes	string	The package dimension code.

Table 5-43 (Cont.) Package Dimension — Object

Element Name	Required	Data Type	Description
weight	Yes	number	The weight of the package in kilograms.
width	Yes	integer(\$int32)	The width of the package in centimeters.

Table 5-44 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions. For example, <pre>{ "en-US": "small", "de-DE": "klein", "fr-FR": "petit" }</pre>

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Update a PackageDimension — PUT

This section describes the Update PackageDimension API. Use this service to update a specific package dimension record's dimensions or locale descriptions.

Method

PUT

URL

/api/foundation/v1/package/dimension/{id}

Allowed Scope(s)

foundation-r
foundation-rw

Request

This section describes the request parameters.

The media type is application/json.

Table 5-45 Update a PackageDimension — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Example Value

```
{
  "created": "2024-05-28T14:04:26.076Z",
  "createdBy": "string",
  "updated": "2024-05-28T14:04:26.076Z",
  "updatedBy": "string",
  "id": <id>,
  "descriptions": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "height": 9999,
  "length": 9999,
  "packDimensionCode": "string",
  "weight": 10000000000000000,
  "width": 9999
}
```

Schema — Package Dimension

This section describes the Package Dimension schema.

Table 5-46 Package Dimension — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
height	Yes	integer(\$int32)	The height of the package in centimeters.
length	Yes	integer(\$int32)	The length of the package in centimeters.
packDimensionCode	Yes	string	The package dimension code.
weight	Yes	number	The weight of the package in kilograms.
width	Yes	integer(\$int32)	The width of the package in centimeters.

Table 5-47 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions. For example, <pre>{ "en-US": "small", "de-DE": "klein", "fr-FR": "petit" }</pre>

Responses

This section describes the responses of the Update a PackageDimension API.

Response Code: 200

The updated record.

The media type is application/json.

Example Value

```
{
  "created": "2024-05-28T14:04:26.163Z",
  "createdBy": "string",
  "updated": "2024-05-28T14:04:26.163Z",
  "updatedBy": "string",
  "id": <id>,
  "descriptions": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "height": 9999,
  "length": 9999,
  "packDimensionCode": "string",
  "weight": 10000000000000000,
  "width": 9999
}
```

Schema — Package Dimension

This section describes the Package Dimension schema.

Table 5-48 Package Dimension — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.

Table 5-48 (Cont.) Package Dimension — Object

Element Name	Required	Data Type	Description
height	Yes	integer(\$int32)	The height of the package in centimeters.
length	Yes	integer(\$int32)	The length of the package in centimeters.
packDimensionCode	Yes	string	The package dimension code.
weight	Yes	number	The weight of the package in kilograms.
width	Yes	integer(\$int32)	The width of the package in centimeters.

Table 5-49 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions. For example, { "en-US": "small", "de-DE": "klein", "fr-FR": "petit" }

Reason Code: 400

The specified ID is not consistent with the provided record, or the provided record is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
```



```
    "errorType": "INTEGRATION"  
  }  
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 5-50 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 5-51 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 404

The provided record could not be found to update.

Reason Code: 409

A related record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Delete a PackageDimension — DELETE

This section describes the Delete a PackageDimension API. This service deletes a package dimension code and all of the associated dimensions and descriptions configured by locale.

Method

DELETE

URL

/api/foundation/v1/package/dimension/{id}

Allowed Scope(s)

foundation-r
foundation-rw

Request

This section describes the request parameters.

Table 5-52 Delete a PackageDimension— Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID

Responses

This section describes the responses of the Delete a PackageDimension API.

Reason Code: 204

Deletion succeeded.

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Checks if a PackageDimension Exists — HEAD

This section describes the Checks if a PackageDimensionn Exists API.

Method

HEAD

URL

/api/foundation/v1/package/dimension/{id}

Allowed Scope(s)

foundation-r

Request

This section describes the request parameters.

Table 5-53 Checks if a PackageDimension Exists — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID

Responses

This section describes the responses of the Checks if a PackageDimension Exists API.

Reason Code: 200

The requested record exists.

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for PackageDimension Records by Attributes — GET

This section describes the Queries for PackageDimension Records by Attributes API. This API returns a list of records matching the provided attributes.

Method

GET

URL

/api/foundation/v1/package/dimension/byattributes

Allowed Scope(s)

foundation-r

Request

This section describes the request parameters.

Table 5-54 Queries for PackageDimension Records by Attributes — Request Parameters

Parameter	Required	Data Type	Description
id	NA	integer(\$int64) (query)	id
height	NA	integer(\$int32) (query)	height
length	NA	integer(\$int32) (query)	length
packDimensionCode	NA	string (query)	packDimensionCode
weight	NA	number (query)	weight
width	NA	integer(\$int32) (query)	width
orderBy	NA	string (query)	A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by " asc " (default) or " desc " to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.

Table 5-54 (Cont.) Queries for PackageDimension Records by Attributes — Request Parameters

Parameter	Required	Data Type	Description
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.
limit	NA	integer(\$int32) (query)	Provides a maximum number of records to be returned.
offset	NA	integer(\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for PackageDimension Records by Attributes API.

Response Code: 200

The requested record.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-07-24T16:05:15.787Z",
    "createdBy": "string",
    "updated": "2024-07-24T16:05:15.787Z",
    "updatedBy": "string",
    "id": <id>,
    "address1": "string",
    "address2": "string",
    "address3": "string",
    "address4": "string",
    "addressType": "RESIDENTIAL",
    "apartment": "string",
    "city": "string",
    "country": "string",
    "deliveryInstructions": "string",
    "emailAddress": "string",
    "firstName": "string",
    "lastName": "string",
    "latitude": 999999999,
```

```

        "locTypeDescription": "string",
        "locationCode": "string",
        "locationDesc": "string",
        "longitude": 999999999,
        "operatingHours": "string",
        "phoneNumber": "string",
        "postalCode": "string",
        "state": "string",
        "timezone": "string"
    }
]

```

Schema — Package Dimension

This section describes the Package Dimension schema.

Table 5-55 Package Dimension — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
height	Yes	integer(\$int32)	The height of the package in centimeters.
length	Yes	integer(\$int32)	The length of the package in centimeters.
packDimensionCode	Yes	string	The package dimension code.
weight	Yes	number	The weight of the package in kilograms.
width	Yes	integer(\$int32)	The width of the package in centimeters.

Table 5-56 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions. For example, <pre>{ "en-US": "small", "de-DE": "klein", "fr-FR": "petit" }</pre>

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Reason Code Service

This section describes the Reason Code Service.

**Note:**

The Reason Code Service can be used to add, replace, edit or delete the default values.

Add a New Reason — POST

This section describes the Add a New Reason API. This service creates a reason code which associates a reason type to a reason description that can be configured by locale. This is also leveraged by the Collect and Receive user interface to allow users to configure a list of reasons for canceling a delivery.

Method

POST

URL

/api/foundation/v1/reason

Allowed Scope(s)

foundation-r
foundation-rw

Request

This section describes the request parameters.

The request body is application/json.

The ModelField is null.

Table 5-57 Add a New Reason— Request Parameters

Parameter	Required	Data Type	Description
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Example Value

```
{
  "created": "2024-06-03T12:13:49.444Z",
  "createdBy": "string",
  "updated": "2024-06-03T12:13:49.444Z",
  "updatedBy": "string",
  "id": <id>,
  "descriptions": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "reasonCode": "string",
  "reasonType": "CANCEL"
}
```

Schema — Reason

Table 5-58 Reason — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
reasonCode	Yes	string(15)	The unique code for the reason.
reasonType	Yes	string	The reason type as to why the action is needed. ENUM: CANCEL

Table 5-59 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions.

Responses

This section describes the responses of the Add a New Reason API.

Response Code: 200

The new record.

The media type is application/json.

Example Value

```
{
  "created": "2024-07-24T06:08:23.004Z",
  "createdBy": "string",
  "updated": "2024-07-24T06:08:23.004Z",
  "updatedBy": "string",
  "id": <id>,
  "idsId": "string",
```

```
"locale": "en-US",  
"preferredLocationCode": "string"  
}
```

Schema — Reason

Table 5-60 Reason — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
reasonCode	Yes	string(15)	The unique code for the reason.
reasonType	Yes	string	The reason type as to why the action is needed. ENUM: CANCEL

Table 5-61 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions.

Reason Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[  
  {  
    "column": 0,  
    "line": 0,  
    "errorCode": 0,  
    "errorMessage": "string",  
  }  
]
```

```
    "moreInfo": {  
      "code": "string",  
      "description": "string"  
    },  
    "field": "string",  
    "value": "string",  
    "errorType": "INTEGRATION"  
  }  
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 5-62 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 5-63 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 409

The record already exists or a related entity was not found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Reason Records— GET

This section describes the Queries for Reason Records API. This service returns a list of reason code records with associated reason type and descriptions matching the provided criteria.

Method

GET

URL

`/api/foundation/v1/reason`

Allowed Scope(s)

`foundation-r`

Request

This section describes the request parameters.

Table 5-64 Queries for Reason Records— Request Parameters

Parameter	Required	Data Type	Description
Where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, that is, a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example, {"child.name":"Jo"} would match Joe, John, and so on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example, {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example, {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is, as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is, "@and1", "@and2", and so on.) to ensure that they are uniquely named since any duplicates will be eliminated.</p>

Table 5-64 (Cont.) Queries for Reason Records— Request Parameters

Parameter	Required	Data Type	Description
orderBy	NA	string (query)	A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by " asc " (default) or " desc " to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.
limit	NA	integer(\$int32) (query)	Provides a maximum number of records to be returned.
offset	NA	integer(\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of the Queries for Reason Records API.

Response Code: 200

The requested record.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-05-29T11:38:19.704Z",
    "createdBy": "string",
    "updated": "2024-05-29T11:38:19.704Z",
    "updatedBy": "string",
    "id": <id>,
    "descriptions": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "reasonCode": "string",
```

```
        "reasonType": "string"  
    }  
]
```

Schema — Reason

Table 5-65 Reason — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
reasonCode	Yes	string(15)	The unique code for the reason.
reasonType	Yes	string	The reason type as to why the action is needed. ENUM: CANCEL

Table 5-66 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Performs a Bulk Action Using a Collection of Reason — GET

This section describes the Performs a Bulk Action Using a Collection of Reason API. This service performs a bulk action (Add, Add or Update, Delete, Update) using a collection of Reason services.

Method

POST

URL

/api/foundation/v1/reason/bulk

Allowed Scope(s)

foundation-r

Request

This section describes the request parameters.

The request body is application/json.

A list of records to be loaded.

Table 5-67 Performs a Bulk Action Using a Collection of Reason — Request Parameters

Parameter	Required	Data Type	Description
action	NA	string (query)	Provides an action to be used performing the bulk operation. If not specified, the default of ADD_OR_UPDATE is used. Available values : ADD, ADD_OR_UPDATE, DELETE, UPDATE

Example Value

```
[
  {
    "created": "2024-05-29T12:03:45.657Z",
    "createdBy": "string",
    "updated": "2024-05-29T12:03:45.657Z",
    "updatedBy": "string",
    "id": <id>,
    "descriptions": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "reasonCode": "string",
    "reasonType": "string"
  }
]
```



```
}  
]
```

Schema — Reason

Table 5-68 Reason — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
reasonCode	Yes	string(15)	The unique code for the reason.
reasonType	Yes	string	The reason type as to why the action is needed. ENUM: CANCEL

Table 5-69 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions.

Responses

This section describes the responses of the Performs a Bulk Action using a Collection of Reason API.

Response Code: 200

The result of the loading operation.

The media type is application/json.

Example Value

```
{
  "errorRecords": [
    {
      "index": 0,
      "error": "CONFLICT"
    }
  ],
  "successCount": 0
}
```

Schema — Bulk Operation Result

The following table describes the elements of a Bulk Operation, describing the result of a bulk operation.

Table 5-70 Bulk Operation Result — Object

Element Name	Required	Data Type	Description
errorRecords	NA	object	A list of ErrorRecord objects describing any records which were not successfully processed.
successCount	NA	integer(\$int32)	The count of the records which were successfully processed.

The following table describes an object describing an error which occurred during a bulk operation.

Table 5-71 Error Record— Object

Element Name	Required	Data Type	Description
index	NA	integer(\$int32)	The 0-based index of the associated record which was not processed.

Table 5-71 (Cont.) Error Record— Object

Element Name	Required	Data Type	Description
error	NA	string	<p>An enumeration describing the result of a bulk action which was not successfully completed.</p> <ul style="list-style-type: none">• CONFLICT - The record was provided with criteria which conflicts with another record.• DOES_NOT_EXIST - The record was provided with an action which required it to exist, but the record was not found.• ALREADY_EXISTS - The record was provided with an action which required it not to exist, but the record was found.• RELATED_DOES_NOT_EXIST - The record was provided without a required related relationship.• INVALID - The record was invalid. Cases for this include:<ul style="list-style-type: none">– The record was provided as an ADD but included an ID.– The record was provided as a DELETE or UPDATE but lacked an ID.– The record was provided with an illegal/invalid field value.• UNKNOWN - Some other unexpected exception occurred. <p>Enum: [CONFLICT, DOES_NOT_EXIST, ALREADY_EXISTS, RELATED_DOES_NOT_EXIST, INVALID, UNKNOWN]</p>

Reason Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
```

```

        "description": "string"
      },
      "field": "string",
      "value": "string",
      "errorType": "INTEGRATION"
    }
  ]

```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 5-72 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 5-73 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 413

Too many records were provided.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Reason Records/Count— GET

This section describes the Queries for Reason Records/Count API. The API returns a count of records matching the provided criteria.

Method

GET

URL

`/api/foundation/v1/reason/count`

Allowed Scope(s)

`foundation-r`

Request

This section describes the request parameters.

Table 5-74 Queries for Reason Records/Count — Request Parameters

Parameter	Required	Data Type	Description
where	NA	object (query)	<p>A JSON object containing query parameters for filtering results. Fields to be filtered on may be provided in the form of a dot-notated path named for the fields known to this resource, that is, a delimited field named "child.id" would filter by the entity field of this resource named "child" and that entity has a field named "id". Querying is supported in 3 ways. If a single value is provided, it is treated as a baseline. If the single value is a character type, then the value is used as the basis for a "like" or startsWith-type comparison, for example, {"child.name":"Jo"} would match Joe, John, and so on. For all other types, it serves as the lower bound of a >= comparison, {"minAmount":100.00}. If an array is provided, the values within it are treated as "equals" or "in" values depending upon the element count, for example, {"id":[1, 2, 3]}. If an object is provided, fields named "from" and "to" within the object, will serve as lower and/or upper bounds for a range-based query, for example, {"id":{"from":0,"to":100}}, a field named "singleValue" can be used to follow the single value pattern discussed above, a field named "in" can be used to follow the array pattern discussed above, or a field named "fieldPath" can be used to specify the path of a related field. Additionally, a field named "caseInsensitive" can be used to enable case-insensitive querying of character-based fields and a field named "invert" can be used to invert a predicate (that is, as a NOT). Combining of criteria in "and" and "or" blocks is supported by providing specially named properties with names beginning with "@and" and "@or" with JSON object values. These can be used directly within the body for criteria grouping purposes, and also nested within each other. When using multiple @and/@or groupings within a given object, take care to suffix them with additional characters (that is, "@and1", "@and2", and so on.) to ensure that they are uniquely named since any duplicates will be eliminated.</p>

Responses

This section describes the responses of the Queries for Reason Records/Count API.

Response Code: 200

The count of requested records.

The media type is application/json.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Get a Reason — GET

This section describes the Get a Reason API. Use this service to retrieve a specific reason record and the associated reason type and descriptions.

Method

GET

URL

/api/foundation/v1/reason/{id}

Allowed Scope(s)

foundation-r

Request

This section describes the request parameters.

Table 5-75 Get a Reason — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Responses

This section describes the responses of the Get a Reason API.

Response Code: 200

The requested record.

The media type is application/json.

Example Value

```
{
  "created": "2024-06-03T13:08:23.275Z",
  "createdBy": "string",
  "updated": "2024-06-03T13:08:23.275Z",
  "updatedBy": "string",
  "id": <id>,
  "descriptions": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "reasonCode": "string",
  "reasonType": "CANCEL"
}
```

Schema — Reason

Table 5-76 Reason — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
reasonCode	Yes	string(15)	The unique code for the reason.
reasonType	Yes	string	The reason type as to why the action is needed. ENUM: CANCEL

Table 5-77 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions.

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Update a Reason — PUT

This section describes the Update a Reason API. Use this service to update a specific reason record's type or locale descriptions.

Method

PUT

URL

`/api/foundation/v1/reason/{id}`

Allowed Scope(s)

foundation-r
foundation-rw

Request

This section describes the request parameters.

The media type is application/json.

Table 5-78 Update a Reason — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.

Example Value

```
{
  "created": "2024-05-29T13:34:47.398Z",
  "createdBy": "string",
  "updated": "2024-05-29T13:34:47.398Z",
  "updatedBy": "string",
  "id": <id>,
  "descriptions": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "reasonCode": "string",
  "reasonType": "string"
}
```

Schema — Reason

Table 5-79 Reason — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
reasonCode	Yes	string(15)	The unique code for the reason.
reasonType	Yes	string	The reason type as to why the action is needed. ENUM: CANCEL

Table 5-80 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions.

Responses

This section describes the responses of the Update a Reason API.

Response Code: 200

The updated record.
The media type is application/json.

Example Value

```
{
  "created": "2024-05-29T13:34:47.403Z",
  "createdBy": "string",
  "updated": "2024-05-29T13:34:47.404Z",
  "updatedBy": "string",
  "id": <id>,
  "descriptions": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "reasonCode": "string",
  "reasonType": "string"
}
```

Schema — Reason

Table 5-81 Reason — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
reasonCode	Yes	string(15)	The unique code for the reason.

Table 5-81 (Cont.) Reason — Object

Element Name	Required	Data Type	Description
reasonType	Yes	string	The reason type as to why the action is needed. ENUM: CANCEL

Table 5-82 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions.

Reason Code: 400

The specified ID is not consistent with the provided record, or the provided record is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 5-83 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known

Table 5-83 (Cont.) Mapper Error Data — Object

Element Name	Required	Data Type	Description
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 5-84 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 404

The provided record could not be found to update.

Reason Code: 409

A related record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Delete a Reason — DELETE

This section describes the Delete a Reason API. This service deletes a reason code and it's associated reason type and descriptions configured by locale.

Method

DELETE

URL

```
/api/foundation/v1/package/dimension/{id}
```

Allowed Scope(s)

```
foundation-r  
foundation-rw
```

Request

This section describes the request parameters.

Table 5-85 Delete a PackageDimension— Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID

Responses

This section describes the responses of the Delete a Reason API.

Reason Code: 204

Deletion succeeded.

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Checks if a Reason Exists — HEAD

This section describes the Checks if a Reason Exists API.

Method

HEAD

URL

```
/api/foundation/v1/reason/{id}
```

Allowed Scope(s)

```
foundation-r
```

Request

This section describes the request parameters.

Table 5-86 Checks if a Reason Exists — Request Parameters

Parameter	Required	Data Type	Description
id	Yes	integer(\$int64) (path)	ID

Responses

This section describes the responses of the Checks if a Reason Exists API.

Reason Code: 200

The requested record exists.

Reason Code: 404

The record could not be found.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

Queries for Reason Records by Attributes — GET

This section describes the Queries for Reason Records by Attributes API.

Method

```
GET
```

URL

/api/foundation/v1/reason/byattributes

Allowed Scope(s)

foundation-r

Request

This section describes the request parameters.

Table 5-87 Queries for Reason Records by Attributes — Request Parameters

Parameter	Required	Data Type	Description
id	NA	integer(\$int64) (query)	id
reasonCode	NA	string (query)	reasonCode
reasonType	NA	string (query)	reasonType, Available value: CANCEL
orderBy	NA	string (query)	A comma-delimited list of fields used to determine result ordering or a JSON array of complex sorting criteria. When supplied as strings, names are provided in the same format as the where argument - optionally followed by " asc " (default) or " desc " to control order direction. When supplied as a JSON array, a String fieldName field must be included in the object. Optional String order (asc or desc) and boolean caseInsensitive fields may also be provided.
expands	NA	string (query)	A comma-delimited list of relationship paths, in dot-notated path form (that is, " child.line " would return the relationship named " child ", and its child relationship named " line "), on the resulting model which should be included in the response or " all " if all associated relationships should be included.
limit	NA	integer(\$int32) (query)	Provides a maximum number of records to be returned.
offset	NA	integer(\$int32) (query)	Provides the zero-based value for the position of the first value to be returned.

Responses

This section describes the responses of Queries for Reason Records by Attributes the API.

Response Code: 200

The requested record.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-06-03T08:23:26.454Z",
    "createdBy": "string",
    "updated": "2024-06-03T08:23:26.455Z",
    "updatedBy": "string",
    "id": <id>,
    "descriptions": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "reasonCode": "string",
    "reasonType": "string"
  }
]
```

Schema — Reason

Table 5-88 Reason — Object

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	id
descriptions	Yes	object	Locale types as IETF BCP47 language tags and their descriptions.
reasonCode	Yes	string(15)	The unique code for the reason.
reasonType	Yes	string	The reason type as to why the action is needed. ENUM: CANCEL

Table 5-89 Descriptions — Object

Element Name	Required	Data Type	Description
description	NA	string	Locale types as IETF BCP47 language tags and their descriptions.

Reason Code: 500

An error occurred during persistence or the request failed for some other reason.

6

User Service

This section describes the User Service.

The User Service contains the following OpenAPI endpoint, which can be used to obtain the specification for the services it provides.

```
api/user/openapi
```

currentUser — GET

This section describes the currentUser API. This service retrieves the current user details which is most commonly leveraged by the Collect and Receive user interface.

Method

GET

URL

```
/api/user/v1/user/current
```

Request

There are no request parameters.

Responses

This section describes the responses of the currentUser API.

Response Code: 200

The current user.

The media type is application/json.

Example Value

```
{
  "created": "2024-07-24T06:08:23.004Z",
  "createdBy": "string",
```

```
"updated": "2024-07-24T06:08:23.004Z",
"updatedBy": "string",
"id": <id>,
"idcsId": "string",
"locale": "en-US",
"preferredLocationCode": "string"
}
```

User — Schema

Table 6-1 User

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
idcsId	Yes	string(255)	The user's IDCS id.
locale	NA	string(20)	The locale of the package dimension description. Default: en-US
preferredLocationCode	NA	string(10)	The user's preferred location code.

Reason Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

```
}  
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 6-2 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 6-3 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 403

The user is not authorized to perform the action.

Reason Code: 500

The request failed.

updateUser — PUT

This section describes the updateUser API. This service updates the current user details which is most commonly leveraged by the Collect and Receive user interface.

Method

PUT

URL

/api/user/v1/user

Request

There are no request parameters.

The request body is application/json.

The ModelField request.

Example

```
{
  "created": "2024-07-24T06:17:25.424Z",
  "createdBy": "string",
  "updated": "2024-07-24T06:17:25.424Z",
  "updatedBy": "string",
  "id": <id>,
  "idcsId": "string",
  "locale": "en-US",
  "preferredLocationCode": "string"
}
```

User — Schema

Table 6-4 User

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	The timestamp at which the record was created.
createdBy	NA	string	The identity of the user who created the record.
updated	NA	string(\$date-time)	The timestamp at which the record was last updated.
updatedBy	NA	string	The identity of the user who last updated the record.
id	NA	integer(\$int64)	ID
idcsId	Yes	string(255)	The user's IDCS id.

Table 6-4 (Cont.) User

Element Name	Required	Data Type	Description
locale	NA	string(20)	The locale of the package dimension description. Default: en-US
preferredLocationCode	NA	string(10)	The user's preferred location code.

Responses

This section describes the responses of the updateUser API.

Response Code: 204

The request was processed successfully.

Reason Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 6-5 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 6-6 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.
description	NA	string	An unlocalizable description describing the error.

Reason Code: 500

The request failed.

7

Administration Service

This chapter describes the Administration Service.

createClient — POST

This section describes the createClient endpoint. This service creates a new application client that can be used to call CaR services from another application.

Method

POST

URL

/api/admin/v1/client

Allowed Scope(s)

admin-rw

Request

There are no request parameters.

The request body is application/json.

The ModelField request.

Example Value

```
{
  "scopes": [
    "string"
  ],
  "appDescription": "string"
}
```

Schema — Client Request

The following table describes details of the new client.

Table 7-1 Client Request — Object

Element Name	Required	Data Type	Description
scopes	NA	string	A collection of scopes assignment.
admin	NA	boolean	Indicates if the client is an administrator.
appDescription	NA	string	App description

Responses

This section describes the responses of the createClient endpoint.

Response Code: 200

The result object.

The media type is application/json.

Example Value

```
{
  "scopes": [
    "string"
  ],
  "clientId": "string",
  "clientSecret": "string"
}
```

Schema — Client Response

The following table describes the elements of a response to the create client request.

Table 7-2 Client Response — Object

Element Name	Required	Data Type	Description
scopes	NA	string	A collection of assigned scopes.
clientId	NA	string	Client Id
clientSecret	NA	string	Client secret

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

The table below describes the elements of the Mapper Error Data object.

Table 7-3 Mapper Error Data — Object

Element Name	Required	Data Type	Description
column	NA	integer <int32>	The JSON file column number the error occurred on if known
line	NA	integer <int32>	The JSON file line number the error occurred on if known
errorCode	NA	integer <int32>	A numeric identifier for the type of error
errorMessage	NA	string	The unlocalized error message
moreInfo	NA	object	Additional information pertaining to the error
field	NA	string	The field. This is the last part of the path
value	NA	string	The JSON value that caused the error if known
errorType	NA	string	enum: INTEGRATION, INTERNAL, MAPPING, PARSING, VALIDATION An enumeration of specific known error types.

The following table describes the elements of the additional information pertaining to the error.

Table 7-4 More Info — Object

Element Name	Required	Data Type	Description
code	NA	string	A code describing the error.

Table 7-4 (Cont.) More Info — Object

Element Name	Required	Data Type	Description
description	NA	string	An unlocalizable description describing the error.

Response Code: 424

Cannot generate new client.

Response Code: 500

The request failed.

forgetPrivateData — POST

This section describes the forgetPrivateData endpoint. Use this service to validate if data is eligible to be anonymized. This includes address information, name, phone, and e-mail. Additionally, use this service to anonymize all personal data stored in the database for the parameters specified.

Method

POST

URL

/api/admin/v1/privacy

Allowed Scope(s)

private-rw

Request

This section describes the request parameters.

Table 7-5 forgetPrivateData — Request Parameters

Parameter	Required	Data Type	Description
identifier	Yes	string (query)	The type of private data identifier. Available values : CONTACT_REFERENCE, EMAIL_ADDRESS, LOCATION, PHONE_NUMBER
value	Yes	string (query)	The value corresponding to the private data identifier.
checkOnly	Yes	boolean (query)	An indication that the request should verify the ability to delete, but not delete data.

Responses

This section describes the responses of the forgetPrivateData endpoint.

Response Code: 200

The number of records affected, or to be affected if checkOnly is true, by the call.

The media type is application/json.

Response Code: 204

No records were found matching the supplied criteria.

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

```
}  
]
```

Schema — Mapper Error Data

For more information about the schema Mapper Error Data, see the [Schema — Mapper Error Data](#) section.

Response Code: 409

Dependent data exists which prevents the request from being completed.

Response Code: 500

The request failed.

getPrivateData — GET

This section describes the getPrivateData endpoint. This service retrieves the PII data based on the parameters used in the request.

Method

GET

URL

/api/admin/v1/privacy

Allowed Scope(s)

private-rw

Request

This section describes the request parameters.

Table 7-6 getPrivateData — Request Parameters

Parameter	Required	Data Type	Description
identifier	Yes	string (query)	The type of private data identifier. Available values : CONTACT_REFERENCE, EMAIL_ADDRESS, LOCATION, PHONE_NUMBER
value	Yes	string (query)	The value corresponding to the private data identifier.

Responses

This section describes the responses of the getPrivateData endpoint.

Response Code: 200

The result object.

The media type is application/json.

Example Value

```
[
  {
    "address1": "string",
    "address2": "string",
    "address3": "string",
    "address4": "string",
    "apartment": "string",
    "businessName": "string",
    "city": "string",
    "contactReference": "string",
    "country": "string",
    "emailAddress": "string",
    "firstName": "string",
    "lastName": "string",
    "latitude": 999999999,
    "longitude": 999999999,
    "middleName": "string",
    "phoneNumber": "string",
    "postalCode": "string",
    "state": "string",
    "suffix": "string",
    "usages": [
      "DROPOFF"
    ]
  }
]
```

Schema — Private Data Result

The following table describes the elements of the result data for a private data request.

Table 7-7 Private Data Result — Object

Element Name	Required	Data Type	Description
address1	NA	string(254)	The address line 1.
address2	NA	string(254)	The address line 2.
address3	NA	string(254)	The address line 3.
address4	NA	string(254)	The address line 4.
apartment	NA	string(254)	The apartment
businessName	NA	string(254)	The business name.
city	NA	string(254)	The city or town.
contactReference	NA	string(64)	A reference number for the contact.
country	NA	string(254)	The ISO-3166 country code. Applications can decide whether to use 2 or 3 char codes.
emailAddress	NA	string(254)	The email address.
firstName	NA	string(254)	The first name (given) name.
lastName	NA	string(254)	The last name (family) name.
latitude	NA	number	The latitude of the address.
longitude	NA	number	The longitude of the address.
middleName	NA	string(254)	The middle name.
phoneNumber	NA	string(50)	The phone number.
postalCode	NA	string(20)	The postal code.
state	NA	string(254)	The ISO 3166 state, province or territory code.
suffix	NA	string(254)	A name suffix.
usages	NA	string	The location(s) in which the data was found to be in-use. Enum: DROPOFF, PICKUP, LOCATION

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
```



```
        "description": "string"
      },
      "field": "string",
      "value": "string",
      "errorType": "INTEGRATION"
    }
  ]
}
```

Schema — Mapper Error Data

For more information about the schema Mapper Error Data, see the [Schema — Mapper Error Data](#) section.

Response Code: 500

The request failed.

addScopes — PUT

This section describes the addScopes endpoint. This API adds a set of scopes to an IDCS client.

Method

PUT

URL

/api/admin/v1/scope/{appName}

Allowed Scope(s)

admin-rw

Request

This section describes the request parameters.

Table 7-8 addScopes— Request Parameters

Parameter	Required	Data Type	Description
appName	Yes	string (path)	The IDCS client's display name.

Responses

This section describes the responses of the addScopes endpoint.

Response Code: 204

The request was successful.

Response Code: 400

The provided input is invalid.

The media type is application/json.

Example Value

```
[
  {
    "column": 0,
    "line": 0,
    "errorCode": 0,
    "errorMessage": "string",
    "moreInfo": {
      "code": "string",
      "description": "string"
    },
    "field": "string",
    "value": "string",
    "errorType": "INTEGRATION"
  }
]
```

Schema — Mapper Error Data

For more information about the schema Mapper Error Data, see the [Schema — Mapper Error Data](#) section.

Response Code: 404

IDCS client not found.

Response Code: 424

IDCS request failed.

Response Code: 500

The request failed.

getClients— GET

This section describes the getClients endpoint. This API obtains a list of IDCS clients with a CaR scope for the tenant environment.

Method

GET

URL

/api/admin/v1/scope/client

Allowed Scope(s)

admin-r

Request

There are no request parameters.

Responses

This section describes the responses of the getClients endpoint.

Response Code: 200

List of IDCS clients.

The media type is application/json.

Example Value

```
[
  {
    "created": "2024-11-28T14:46:46.687Z",
    "grantedScopes": [
      "string"
    ],
    "id": "string",
    "name": "string",
    "updated": "2024-11-28T14:46:46.687Z"
```

```
}  
]
```

Schema — Client

Table 7-9 — Client

Element Name	Required	Data Type	Description
created	NA	string(\$date-time)	Date time when the client was created.
grantedScopes	NA	uniqueItems: true string	Granted scopes to the client.
id	NA	string	The IDCS client identifier.
name	NA	string	The IDCS client name.
updated	NA	string(\$date-time)	Date time when the client was last updated.

Response Code: 404

No CaR clients found.

Response Code: 424

IDCS request failed.

Response Code: 500

The request failed.

8

Version Updates

Software updates are critical to keeping an environment secure and functioning well. Critical patch updates are installed on a quarterly basis, for example to the database and other tools being used in Collect and Receive. These updates may require downtime. If this is the case, the planned downtime is communicated in advance according to Oracle standards.