

Envio 360[®] Orchestration

API

Document v2.3
August 2024



Table of Contents

Overview.....	8
Supported Communication Channels.....	8
Supported formats	8
API Specification	9
Base URL	9
Authentication	9
Responses	9
Response codes.....	9
Orchestration endpoints.....	9
Creating orchestrations	9
<i>Examples</i>	17
Updating an orchestration.....	24
Canceling an Orchestration	26
Canceling an Appointment	26
Rescheduling an Appointment.....	27
SCACless appointments	29
Empty Return.....	30
Dual Moves.....	31
Orchestration Status.....	36
Container Status.....	37
Orchestration Status for Empty Returns	47
Orchestration Status for Off-Dock Locations.....	54
Orchestration Status for Dual Matching	57
Receiving notifications.....	61
Notification types	62
Use Cases	63
Invalid request	63
Invalid or missing required field	63
Unit not found.....	63
Orchestration created with warnings	63
Orchestration stopped	64
Non-matching bill of lading	64
Third-Party Fees	65

Appointments not supported.....	66
Appointment canceled by external action	70
Error codes and severity levels	72
Container Status Codes	75
Container Activity Timeline Codes	75
Hold Reason Codes.....	75
Reservation Status.....	76
Flows.....	76
Only Third-Party Payments.....	76

Document Control (amendment history & version control)			
Version	Date	Amended By	Reason
1.0	11/15/22	Envio360	<ul style="list-style-type: none"> Initial version
1.1	01/09/23	Envio360	<ul style="list-style-type: none"> Minor discrepancies fixed
1.2	01/31/23	Envio360	<ul style="list-style-type: none"> Expanded examples and code lookup tables.
1.3	02/07/23	Envio360	<ul style="list-style-type: none"> Expanded use cases
1.4	03/07/23	Envio360	<ul style="list-style-type: none"> Added reschedule use case. Added status notification content example. Minor discrepancies fixed.
1.5	05/10/23	Envio360	<ul style="list-style-type: none"> Expanded error codes: Container not found. Terminal not supported. Payments not supported. Added 'Flows' section.
1.6	07/10/23	Envio360	<ul style="list-style-type: none"> Trucker key replaced by trucker SCAC. Introduced threshold ("lead time"). Added gate code definition. Added notification type definition.
1.7	07/10/23	Envio360	<ul style="list-style-type: none"> Updated code 10004. Updated code 10026. Updated code 10029. Added code 10029. Updated code 10031. Added orchestration stopped event. Added "Unit not found" use case. Added "Orchestration stopped" use case. Added Reschedule definition with and without SCAC. Added definition of reservation preferences in orchestration creation request body. Renamed 'loc_type' to 'location_type'.

1.8	08/07/23	Envio360	<ul style="list-style-type: none"> • Updated code 10020. • Updated code 10032. • Added code 10033. • Fixed “trucker_nbr” as “truck_nbr” in orchestration objects fields description.
1.9	09/07/23	Envio360	<ul style="list-style-type: none"> • Updated “trucker_info” information in “Orchestration fields” • Updated “Reservation Status” table. • Expanded ‘threshold’ definition. • Added details for all errors and warning codes. • Added 10025 error example to “Appointments not Supported” section. • Added “Notification Types” section under “Receiving Notifications”. • Added ‘CMP’ value and updated ‘ACT’ description for ‘locationstatus_cd’ in container status. • Updated “Overview” to reflect changes in orchestration end. • Updated “Orchestration stopped” section to reflect changes in orchestration end. • Removed deprecated “event_cd” and “event_cat” fields in container status. • Updated Empty Return section. • Expanded ‘locationstatus_cd’ definition. • Added ‘Orchestration Status for Empty Returns’ section. • Updated terminal name in Orchestration Status examples. • Added off-dock notification code. • Added off-dock example. • Renamed locations and companies in examples. • Added 1035, 1036 and 1037 error codes.

2.0	10/19/2023	Envio360	<ul style="list-style-type: none"> • Fixed PierPass fee fields in 'fee_receipts' section in Container Status and Only Third Party Payments sections. • Added "Creating an orchestration for Empty Return" section. • Added "Empty Hunting" section.
2.1	01/25/2024	Envio360	<ul style="list-style-type: none"> • Added description in section "Dual Moves". • Added description for error code 10035. • Added description for error code 10036. • Added description for error code 10037. • Added 10038 and 10039 error codes. • Added information to "Empty Hunt" section.
2.2	03/08/2024	Envio360	<ul style="list-style-type: none"> • Added HTTP 429 status code in "API responses" section. • Changed HTTP 202 description to ACCEPTED in "API responses" section. • Added "location_type" field to all orchestration creation example to distinguish between pick import ("O") and empty return ("D"). • Removed not supported "T" (Trailer) option from "Orchestration fields" section. • Replaced "0" by "O" in PATCH /orchestrations/trade-type/l/unit-nbr/ABCD1234567/locations/ • Improved hold examples in "GET /orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/unit-status" section. • Removed orchestration status specific message when an external cancelation is received in "Appointment canceled by external action" section. • Added example on how to retrieve notifications from ASB in "Receiving notifications" section. • Added "sections_updated" field to orchestration notifications.

2.3	08/08/2024	Envio360	<ul style="list-style-type: none"> • Removed incorrect “sztypeiso_cd” and “sztype_cd” fields from “Orchestration fields” • Added “free_time_expiration” field to “Orchestration fields” • Added “dual” section to “Orchestration fields” • Added ‘Dual Matching’ functionality description in “Dual Moves” section. • Added examples for Dual Matching under “Orchestrating a Dual Move by using the ‘dual’ section” • Added “Orchestration Status for Dual Matching” section
-----	------------	----------	---

Overview

Envio360 Orchestration API provides the ability for BCOs to submit container appointment and/or payment orchestration requests and receive status tracing on the submitted containers.

A single request may contain multiple orchestrations to be processed by the Envio360 system, following this flow:

1. Customer sends Orchestration Request (possibly containing multiple orchestrations).
2. Orchestration(s) begins for each container once the container is reported by the terminal. Customer receives container status updates throughout the lifecycle of the orchestration published to Azure Service Bus. The status can also be queried at any time via the REST API.
3. If customer includes payment options, the orchestration keeps fees paid up until the guaranteed through date. A status update is sent to Azure Service Bus on each payment event.
4. If the customer includes appointment preferences, the orchestration requests an appointment as soon as the container is available for it. A status update is sent to Azure Service Bus on each appointment event.
5. The orchestration stays active and can be updated with empty drop appointment preferences.
6. If the customer includes dual move information, the appointment is set up to support both the pick and drop moves.
7. The orchestration ends when the container is gated in.

When creating multiple orchestrations in one API call, we recommend at most 100 orchestrations per API request in order to avoid a possible timeout. Rate limits may apply.

Supported Communication Channels

- REST API integration:
 - Envio360 provides: API Endpoint, Subscription keys
- Microsoft Azure Subscription Endpoints:
 - Envio360 provides: Azure Subscription Key and Subscription endpoints.

Supported formats

- JSON Format:

JSON is the only message format supported to receive orchestration requests and to provide status updates.

API Specification

Please see the specification details below

Base URL

<https://api.envio360.emodal.com/v1>

Authentication

Authentication key is sent in the request header:

Key	Value
Content-Type	application/json
X-API-Key	[[subscription key]]

Responses

Response codes

Code	Status	Reason
200	OK	Resource obtained successfully
202	ACCEPTED	Request accepted successfully
204	NO CONTENT	Resource updated successfully
400	BAD REQUEST	Message schema is invalid/HTTP request invalid
401	UNAUTHORIZED	Authorization Failure
404	NOT FOUND	Resource not found
412	PRECONDITION FAILED	Validation failed
429	TOO MANY REQUESTS	API rate limit exceeded
500	INTERNAL SERVER ERROR	Internal server error
503	SERVICE UNAVAILABLE	The server is currently unable to handle the request due to a temporary overload or scheduled maintenance

Orchestration endpoints

Creating orchestrations

Submit one or many orchestration requests.

URL

POST /orchestrations

Header

See Authentication

Body

See "Create Orchestrations Object" section for details

```
{
  "order": [
    {
      "trade_type": "string",
      "billoflading": "string",
      "unit": {
        "unit_nbr": "string",
        "unit_category": "string",
        "line_scac": "string",
        "sztypiso": "string",
        "seal_nbr": [
          "string"
        ]
      }
    },
    "locations": [
      {
        "port_cd": "string",
        "location_cd": "string",
        "location_type": "string",
        "guarantee_through_date": "string",
        "reservation_details": {
          "free_time_expiration": "string",
          "reservation_preferences": {
            "try_later_dates": "string",
            "preference_datetimes": [
              {
                "range": ["string"],
                "days": ["integer"],
                "after": "string",
                "before": "string"
              }
            ]
          },
          "dual": {
            "match": "boolean",
            "match_cutoff": "integer",
            "primary_match_cutoff": "string",
            "match_preferences": "string"
          }
        },
        "pin_nbr": "string",
        "dual_unit": {
          "unit_info": {
            "unit_nbr": "string",
            "unit_category": "string",
            "line_scac": "string",
            "sztypiso": "string",
            "seal_nbr": [
              "string"
            ]
          }
        }
      }
    ]
  }
}
```

```

    },
    "hazmat_info": {
      "haz_class": "string",
      "haz_unicode": "string",
      "commodity_weight": "string",
      "quantity": "string",
      "package_group": "string",
      "commodity_name": "string",
      "emergency_ph": "string",
      "limited_qty": "string",
      "comments": "string"
    }
  },
  "hazmat_info": {
    "haz_class": "string",
    "haz_unicode": "string",
    "commodity_weight": "string",
    "quantity": "string",
    "package_group": "string",
    "commodity_name": "string",
    "emergency_ph": "string",
    "limited_qty": "string",
    "comments": "string"
  },
  "trucker_info": {
    "trucker_scac": "string",
    "truck_nbr": "string",
    "lic_plate_nbr": "string",
    "lic_plate_state_cd": "string",
    "driver_id": "string",
    "driver_fname": "string",
    "driver_lname": "string"
  }
}
]
}
]
}
}

```

Responses

If multiple orchestrations are submitted, HTTP 202 will be returned if at least one orchestration is accepted. Each orchestration request status will be provided in the data array. HTTP 412 will be returned if all orchestrations in the request fail.

```

{
  "data": [
    {
      "success": "boolean",
      "unit_nbr": "string",
      "messages": [
        {
          "message_cd": "string",

```

```

        "message_txt": "string",
        "message_severity_cd": "string",
    }
  ]
}

```

Name	Description
data	<i>list of object</i> containing response information
data.success	<i>string</i> Success flag associated with the orchestrated container
data.unit_nbr	<i>string</i> Orchestrated container number
data.messages	<i>list of object</i> Messages associated to the container orchestration
data.messages.message_cd	<i>string</i> Message codes associated to the container orchestration
data.messages.message_txt	<i>string</i> Message description associated to the container orchestration
data.messages.message_severity_cd	<i>string</i> Message severity code associated to the container orchestration. I = Informational W = Warning C = Critical

202 Accepted

```

{
  "data": [
    {
      "success": true,
      "unit_nbr": "ABCD1234567",
      "messages": [
        {
          "message_cd": "10000",
          "message_txt": "Orchestration created successfully",
          "message_severity_cd": "I"
        }
      ]
    }
  ]
}

```

202 Accepted

```

{
  "data": [
    {
      "success": true,
      "unit_nbr": "ABCD1234567",

```

```

    "messages": [
      {
        "message_cd": "10007",
        "message_txt": "Missing Guarantee Through date",
        "message_severity_cd": "W"
      }
    ]
  }
}

```

412 Precondition failed

```

{
  "data": [
    {
      "success": false,
      "unit_nbr": "ABCD1234567",
      "messages": [
        {
          "message_cd": "10003",
          "message_txt": "Invalid Port Code",
          "message_severity_cd": "C"
        }
      ]
    }
  ]
}

```

Create Orchestration Object

Orchestration fields

Name	Description
order. trade_type	<i>string</i> Mandatory Trade Type. I = Import, X = Export
order. billoflading	<i>string</i> Mandatory Must match eModal existing inventory of Bill of Lading Numbers
order. unit	<i>object</i> Mandatory. Unit information
order.unit. sztypeiso	<i>string</i> Conditional. Unit size and ISO type code. Mandatory for Export.
order.unit. unit_nbr	<i>string</i> Conditional. Container# for Import/Export/Empties, Chassis# for Bare Claim. Mandatory for import pickups
order.unit. unit_category	<i>string</i> optional C=Container (default)
order.unit. seal_nbrs	<i>list of string</i> Optional Seal Number Array up to 8 entries
order. locations	<i>object</i> Mandatory.

	Object containing information about the location where the unit is to be orchestrated.
order.locations. location_type	<i>string</i> Conditional. O – Origin, there can be only one Origin location per unit. D – Destination, there can be only one Destination location per unit. Mandatory if multiple 1
order.locations. port_cd	<i>string</i> Mandatory. Port Code
order.locations. location_cd	<i>string</i> Optional. Facility or terminal code.
order.locations. guarantee_through_date	<i>object</i> Optional. Date through which payments will be attempted. Payments will not be processed if this value is not provided. 'YYYY-MM-DD'
order.locations. reservation_details	<i>object</i> Optional. Details about reservation request
order.locations.reservation_details. free_time_expiration	<i>string</i> Optional. Date in 'YYYY-MM-DD' format representing the end of free time. For duals, it also represents a limit to attempt a match. Related to `match_preferences` (see below).
order.locations.reservation_details. reservation_preferences	<i>object</i> Optional. Reservation details
order.locations.reservation_details. reservation_preferences. try_later_dates	<i>string</i> Optional. Y = YES (default), N = NO. Try dates later than the indicated dates if one or more date time ranges are provided.
order.locations.reservation_details. reservation_preferences. threshold	<i>string</i> Optional. Number of minutes elapsed after the appointment is requested guaranteeing that any selected time slot is still valid (“lead time”). If no slots available, the system will retry calculating the threshold from the current time at the next attempt. The threshold “slides” with the current time in this way.
order.locations.reservation_details. reservation_preferences. preference_datetimes	<i>list of object</i> Optional. Time preferences and restrictions for selecting the appointment. All dates and time are relative to the local time zone.
order.locations.reservation_details. reservation_preferences.preference_datetimes. range	<i>list of string</i> Optional. Expected as a list containing two items, reflecting the start and the end of the preferred range. Items can be either datetime, time or null. Date format is 'YYYY-MM-DDTHH:MM'. Time format is 'HH:MM'. If null is provided at either the start or the end, no limit will be imposed as start or end of the range. Both ends of the range cannot be null.
order.locations.reservation_details. reservation_preferences.preference_datetimes. days	<i>string</i> Optional. days of the week to allow appointment schedule attempts. 1 is Monday and 7 is Sunday.
order.locations.reservation_details. reservation_preferences.preference_datetimes. after	<i>string</i> Optional. Date in 'YYYY-MM-DD' format representing the earliest date for the preferred appointment date. Any dates with availability before this will be ignored.
order.locations.reservation_details. reservation_preferences.preference_datetimes. before	<i>string</i> Optional. Date in 'YYYY-MM-DD' format representing the earliest date after the preferred appointment date. Any dates with availability on this date and after will be ignored.

order.locations.reservation_details.reservation_preferences.dual	<i>object</i> Optional. Dual matching information
order.locations.reservation_details.reservation_preferences.dual.match	<i>bool</i> Optional. For a pick import orchestration, attempt to match with an empty return. If the orchestration was originally created with match = true, modifying the orchestration with match = false will stop the matching process only if a match hasn't been completed and the dual is not linked yet. If the dual is complete, cancel the empty return appointment to remove the dual.
order.locations.reservation_details.reservation_preferences.dual.match_cutoff	<i>integer</i> Optional. Number of minutes prior to the pick import appointment start as a limit to the orchestration attempting to match an empty return. If zero or no cutoff minutes are provided, the orchestration will attempt to find an empty return until the start of the appointment time slot considering the threshold imposed for the empty preferences, if any threshold provided. See "threshold" for more information.
order.locations.reservation_details.reservation_preferences.dual.primary_match_cutoff	<i>string</i> Optional. For empty orchestrations, date in 'YYYY-MM-DD' format representing a limit for an "ideal" match. Related to `match_preferences` (see below). After `primary_match_cutoff`, different matching preferences can be applied.
order.locations.reservation_details.reservation_preferences.dual.match_preferences	<i>string</i> Optional. For empty orchestrations, comma separated list of <i>key=value</i> , representing types of matches that are allowed/preferred. Preferences can be passed for the primary window (before `primary_match_cutoff` date) and the secondary window (after `primary_match_cutoff` and until `free_time_expiration` date). key = AD1 (Actual Dual in primary window), VD1 (Virtual Dual in primary window), SA1 (Stand Alone in primary window), AD2 (Actual Dual in secondary window, VD2 (Virtual Dual in secondary window), SA2 (Stand Alone in secondary window) value = 0, 1, 2, 3 with 0 = disable that type of match, and 1, 2, 3, representing the priority for the type of match, with 1 being the highest priority. AD (Actual Dual): match for same terminal. VD (Virtual Dual): match for different terminal or same terminal at a different time SA (Stand Alone): secure the appointment even if no dual could be matched. See examples below.
order.locations.reservation_details.pin_nbr	<i>string</i> Optional. Gate Code that may be required by the terminal.
order.locations.reservation_details.hazmat_info.haz_class	<i>object</i> Optional Hazardous Class.
order.locations.reservation_details.hazmat_info.haz_unicode	<i>string</i> Optional Hazardous UN Code
order.locations.reservation_details.hazmat_info.commodity_weight	<i>string</i> Optional Hazardous commodity weight - This value to be received in Lbs
order.locations.reservation_details.hazmat_info.quantity	<i>string</i> Optional Hazardous quantity

order.locations.reservation_details.hazmat_info. package_group	<i>string</i> Optional Hazardous package group
order.locations.reservation_details.hazmat_info. commodity_name	<i>string</i> Optional Hazardous commodity Name
order.locations.reservation_details.hazmat_info. emergency_ph	<i>string</i> Optional Emergency Phone number for Hazardous
order.locations.reservation_details.hazmat_info. limited_qty	<i>string</i> Optional Indicates Limited Quantity which designates placard is not needed. Values 'Y' 'N' or null
order.locations.reservation_details.hazmat_info. comments	<i>string</i> Optional comments on Hazardous item
order.locations.reservation_details. trucker_info	<i>object</i> Optional trucker information
order.locations.reservation_details.trucker_info. trucker_scac	<i>string</i> Optional Trucker SCAC. If provided, the appointment will be assigned to the trucking company represented by the SCAC. If no SCAC is provided, the appointment will be held SCACless until it's assigned to the trucking company. Required to execute an appointment.
order.locations.reservation_details.trucker_info. truck_nbr	<i>string</i> Optional Truck Number assigned by the Trucking Company. Currently not required to execute an appointment. However, terminals may enforce this fields at any time without prior warning.
order.locations.reservation_details.trucker_info. lic_plate_nbr	<i>string</i> Optional License Plate Number. In addition to 'trucker_scac', this is the only field required by some terminals prior to arriving at the facility. In that case, appointment can be requested without this information and can be updated later including this value.
order.locations.reservation_details.trucker_info. lic_plate_state_cd	<i>string</i> Optional License Plate State Code. Currently not required to execute an appointment. However, terminals may enforce this fields at any time without prior warning.
order.locations.reservation_details.trucker_info. driver_id	<i>string</i> Optional Driver's License number or ID. Currently not required to execute an appointment. However, terminals may enforce this fields at any time without prior warning.
order.locations.reservation_details.trucker_info. driver_fname	<i>string</i> Optional Driver's First Name. Currently not required to execute an appointment. However, terminals may enforce this fields at any time without prior warning.
order.locations.reservation_details.trucker_info. driver_lname	<i>string</i> Optional Driver's Last Name. Currently not required to execute an appointment. However, terminals may enforce this fields at any time without prior warning.

Examples

Create Import Orchestration for trace only

The following request will create an orchestration for trace only (not appointment or payment).

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA"
        }
      ]
    }
  ]
}
```

Create Import Orchestration for payment only

The following request will create an orchestration for payment (but not appointment). Trace information will also be provided (trace is part of the “Base Orchestration”).

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "guarantee_through_date": "2021-01-01"
        }
      ]
    }
  ]
}
```

Create Import Orchestration for the first available reservation

The following request will create an orchestration for payment and the first available pick import appointment (for empty returns please see “Creating an orchestration for Empty Return” section. Trace information will also be provided.

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "location_type": "O",
          "guarantee_through_date": "2021-01-01",
          "reservation_details": {}
        }
      ]
    }
  ]
}
```

The following request will create an orchestration for payment and the first available pick import appointment as well. This time, PIN number is provided upfront.

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "location_type": "O",
          "guarantee_through_date": "2021-01-01",
          "reservation_details": {
            "reservation_preferences": {}
            "pin_nbr": "1234"
          }
        }
      ]
    }
  ]
}
```

Create Import Orchestration for the first available slot with two hours of “lead time”.

The following request will create an orchestration for payment and the first available pick import appointment, guaranteeing that the selected slot will end after the given threshold in minutes. If it's currently 1:30pm, and a slot from 2:00pm to 3:00pm is available, a 120-minute threshold will prevent this slot from being assigned since the threshold falls outside of the window. A later slot will be selected if available.

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "location_type": "O",
          "guarantee_through_date": "2022-01-01",
          "reservation_details": {
            "reservation_preferences": {
              "try_later_dates": "N",
              "threshold": "120"
            }
          }
        }
      ]
    }
  ]
}
```

Using a threshold to search for a slot for a limited time, and fail if no slot found after the limit

The following request will create an orchestration for payment and a pick import appointment at a given date from 6:00pm to 10:00pm on Dec 30th, 2022. If the request is sent at 8:00 am on Dec 29th, 2022, and the request must error out after four hours if no appointment was found, the threshold value needs to be 2040.

Basically, subtract the time limit from the end of the time preferences range: Dec 30th, 2022 10:00pm – Dec 29th, 2022 12:00pm (four hours after submitting the request) = 34 hours = 2040 minutes of threshold.

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
```

```

    "port_cd": "PORTA",
    "location_type": "0",
    "guarantee_through_date": "2022-01-01",
    "reservation_details": {
      "reservation_preferences": {
        "try_later_dates": "N",
        "threshold": "240",
        "preference_datetimes": [
          {
            "range": ["2021-12-30T18:00:00", "2021-12-30T22:00:00"]
          }
        ]
      }
    }
  }
]
}

```

Create Import Orchestration for reservation after a given date and time

The following request will attempt to create a pick import appointment for Jan 1st, 2021 after 7am. If no availability is found, later dates and times will be attempted until the appointment is confirmed.

POST /orchestrations

```

{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "location_type": "0",
          "guarantee_through_date": "2022-01-01",
          "reservation_details": {
            "reservation_preferences": {
              "preference_datetimes": [
                {
                  "range": ["2021-01-01T07:00:00", null]
                }
              ]
            }
          }
        }
      ]
    }
  ]
}

```

```
]
}
```

Create Import Orchestration for reservation on a given date only

The following request will attempt to create a pick import pick import appointment for Jan 1st, 2021 after 7am. If no availability is found, the request will error out.

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "location_type": "O",
          "guarantee_through_date": "2021-01-01",
          "reservation_details": {
            "reservation_preferences": {
              "try_later_dates": "N",
              "preference_datetimes": [
                {
                  "range": ["2021-01-01T07:00:00", null]
                }
              ]
            }
          }
        }
      ]
    }
  ]
}
```

Create Import Orchestration for reservation between dates and times

An appointment can be requested between datetimes. In this example, the client is requesting a pick import appointment between Jan 1st, 2021 at 7am and Jan 7th, 2021 at 5pm.

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
```

```

    "port_cd": "PORTA",
    "location_type": "0",
    "guarantee_through_date": "YYYY-MM-DD",
    "reservation_details": {
      "reservation_preferences": {
        "preference_datetimes": [
          {
            "range": ["2021-01-01T07:00:00", "2021-01-07T17:00:00"]
          }
        ]
      }
    }
  ]
}
]
}

```

Create Import Orchestration for reservation in time ranges

Appointments can also be requested to be confirmed between time ranges. The date will be assumed to be the current date and later dates will be tried if no availability is found on the current date. In this example, “try every day from today, from 6:00 am to 2:00 pm and from 4:30 pm to 6:00 pm”

POST /orchestrations

```

{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "location_type": "0",
          "guarantee_through_date": "YYYY-MM-DD",
          "reservation_details": {
            "reservation_preferences": {
              "preference_datetimes": [
                {
                  "range": ["06:00:00", "14:00:00"]
                },
                {
                  "range": ["16:30:00", "18:00:00"]
                }
              ]
            }
          }
        }
      ]
    }
  ]
}

```

```
]
}
```

Create Import Orchestration for reservation multiple time ranges

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "location_type": "O",
          "guarantee_through_date": "YYYY-MM-DD",
          "reservation_details": {
            "reservation_preferences": {
              "preference_datetimes": [
                {
                  "range": ["06:00:00", "14:00:00"],
                  "days": [2, 4, 6]
                },
                {
                  "range": ["16:30:00", "18:00:00"],
                  "days": [1, 3, 5]
                }
              ]
            }
          }
        }
      ]
    }
  ]
}
```

Create Import Orchestration for reservation multiple time ranges after a given date

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "location_type": "O",
          "guarantee_through_date": "YYYY-MM-DD",
```

```

    "reservation_details": {
      "reservation_preferences": {
        "preference_datetimes": [
          {
            "range": ["06:00:00", "14:00:00"],
            "days": [2, 4, 6],
            "after": "YYYY-MM-DD",
            "before": "YYYY-MM-DD"
          },
          {
            "range": ["16:30:00", "18:00:00"],
            "days": [1, 3, 5],
            "after": "YYYY-MM-DD",
            "before": "YYYY-MM-DD"
          }
        ]
      }
    }
  ]
}

```

Updating an orchestration

After an orchestration is created, it can be updated later with preferences or required information.

Request

PATCH /orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/locations/{location_type}

Header

See Authentication

Body

```

{
  "guarantee_through_date": "string",
  "reservation_details": {
    "reservation_preferences": {
      "trylaterdates": "string",
      "preference_datetimes": [
        ],
      },
    "pin_nbr": "string"
  },
  "trucker_info": {
    "trucker_scac": "string",
    "truck_nbr": "string",
    "lic_plate_nbr": "string",
    "lic_plate_state_cd": "string",
    "driver_id": "string",
    "driver_fname": "string",
    "driver_lname": "string"
  }
}

```

```
}  
}  
}
```

Responses

200 OK

```
{  
  "messages": [  
    {  
      "message_cd": "10029",  
      "message_txt": "Orchestration update request accepted",  
      "message_severity_cd": "I",  
    }  
  ]  
}
```

412 Precondition Failed

```
{  
  "messages": [  
    {  
      "message_cd": "10008",  
      "message_txt": "Invalid preferences: {preferences}",  
      "message_severity_cd": "C"  
    }  
  ]  
}
```

Examples

Update Guarantee Through Date

PATCH /orchestrations/trade-type/l/unit-nbr/ABCD1234567/locations/O

```
{  
  "guarantee_through_date": "2022-01-01"  
}
```

Update an orchestration to schedule a pick import appointment on the first available slot

PATCH /orchestrations/trade-type/l/unit-nbr/ABCD1234567/locations/O

```
{  
  "reservation_details": {}  
}
```

Update trucker information for pick import appointment

PATCH /orchestrations/trade-type/l/unit-nbr/ABCD1234567/locations/O

```
{  
  "reservation_details": {
```

```
"trucker_info": {
  "lic_plate_nbr": "6TRJ244",
  "lic_plate_state_cd": "CA",
  "driver_fname": "J",
  "driver_lname": "Doe"
}
```

Canceling an Orchestration

Request

DELETE /orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}

Header

See Authentication

Body

None

Responses

202 Accepted

```
{
  "messages": [
    {
      "message_cd": "10010",
      "message_txt": "Orchestration canceled successfully",
      "message_severity_cd": "I"
    }
  ]
}
```

404 Not Found

```
{
  "messages": [
    {
      "message_cd": "10009",
      "message_txt": "Orchestration not found",
      "message_severity_cd": "C"
    }
  ]
}
```

Canceling an Appointment

Request

DELETE /orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/locations/{location_type}

Header

See Authentication

Body

None

Responses

200 OK

```
{
  "messages": [
    {
      "message_cd": "10011",
      "message_txt": "Appointment canceled successfully",
      "message_severity_cd": "I",
    }
  ]
}
```

404 Not Found

```
{
  "messages": [
    {
      "message_cd": "10032",
      "message_txt": "Appointment not found in given location",
      "message_severity_cd": "C"
    }
  ]
}
```

Rescheduling an Appointment

Reschedule an appointment by updating the time preferences. If updates to other fields than the time preferences are provided, the request can fail if the appointment is already scheduled as both rescheduling and updating appointment details at the same time is not supported.

Request

PATCH/orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/locations/{location_type}

Header

See Authentication

Body

```
{
  "reservation_details": {
    "reservation_preferences": {
      "trylaterdates": "string",
      "preference_datetimes": [
      ]
    }
  }
}
```

```
}
```

Submit new reservation preferences to trigger a reschedule if the appointment has been scheduled previously. If no appointment has been scheduled, the new time preferences will be used to schedule. Do not include SCAC if appointment is still SCACless (See "SCACless Appointments").

IMPORTANT: Include the trucker SCAC if the appointment has been already assigned to a trucker.

Body

```
{
  "reservation_details": {
    "reservation_preferences": {
      "trylaterdates": "string",
      "preference_datetimes": [
    ]
    },
    "trucker_info": {
      "trucker_scac": "string"
    }
  }
}
```

Responses

202 OK

```
{
  "messages": [
    {
      "message_cd": "10029",
      "message_txt": "Orchestration update request accepted",
      "message_severity_cd": "I",
    }
  ]
}
```

404 Not Found

```
{
  "messages": [
    {
      "message_cd": "10032",
      "message_txt": "Appointment not found in given location",
      "message_severity_cd": "C"
    }
  ]
}
```

412 Conflict

```
{
  "messages": [
    {
      "message_cd": "10033",
      "message_txt": "Appointment cannot be updated while awaiting response from internal appointment system",
    }
  ]
}
```

```

    "message_severity_cd": "C"
  }
]
}

```

The appointment cannot be rescheduled if the previous appointment is PENDING (terminal is processing and has not responded yet). Appointments can be rescheduled only for SCACless (see SCACless appointments) or for confirmed appointments.

412 Conflict

```

{
  "messages": [
    {
      "message_cd": "10031",
      "message_txt": "Updates cannot be passed during a schedule request",
      "message_severity_cd": "C"
    }
  ]
}

```

The appointment cannot be rescheduled while other details (such as Driver's information) are updated.

SCACless appointments

Every call to the orchestration API is authenticated by a subscription key (See authentication). The subscription key represents a company within the system. BCO company types can request an appointment and assign to a trucking company later. This needs to be executed before the appointment time or the slot will be missed.

Initiating a SCACless appointment (first available)

POST /orchestrations

```

{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "location_type": "O",
          "guarantee_through_date": "YYYY-MM-DD",
          "reservation_details": {}
        }
      ]
    }
  ]
}

```

Providing a Trucking Company

PATCH /orchestrations/trade-type/I/unit-nbr/ABCD1234567/locations/O

```
{
  "reservation_details": {
    "trucker_info": {
      "trucker_scac": "SCAC",
      "lic_plate_nbr": "ABCD",
      "lic_plate_state_cd": "CA",
      "driver_fname": "J",
      "driver_lname": "D"
    }
  }
}
```

Trucker details can be added later:

PATCH /orchestrations/trade-type/I/unit-nbr/ABCD1234567/locations/O

```
{
  "reservation_details": {
    "trucker_info": {
      "lic_plate_nbr": "ABCD",
      "lic_plate_state_cd": "CA",
      "driver_fname": "J",
      "driver_lname": "D"
    }
  }
}
```

Empty Return

Empty returns must be added after the container has gated out. Scheduling preferences, assigning trucker information, rescheduling and canceling work in the same fashion as import picks.

Creating an orchestration for Empty Return

The following request will create an orchestration for the first available appointment on an empty container:

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "location_type": "D",
          "reservation_details": {}
        }
      ]
    }
  ]
}
```

```
]
}
```

The reservation details can be passed in the same way as picks. Guarantee Through Date is not required.

If the container is not found or **has not gated out yet**, the system will report a not found error.

Adding an Empty Return to an existing orchestration

This example creates a first available appointment request for the empty return.

```
PATCH /orchestrations/trade-type/I/unit-nbr/ABCD1234567/locations/D
```

```
{
  "port_cd": "PORTA",
  "reservation_details": {}
}
```

Empty Hunting

The system will try to schedule an appointment at the terminal where the pick was performed. However, if this terminal is unavailable for returning the container, the system will try to schedule an appointment matching the schedule preferences at the next terminal within the same port authority. The terminals will be tried in an unspecified order. If no slots found, the system will report the “no time slots found” warning and keep looking.

The container status (See Container Status section) will show the location as “UNKNOWN” with the passed preferences until the location is determined by the system.

Empty returns not supported

If the terminal does not accept empty returns, the following warning will be added to the orchestration status (see sections “Orchestration Status” and “Receiving Notifications”).

- 10029 “Empty return not allowed by terminal”.

Canceling an Empty Return

Empty returns can be canceled regardless of the status of the underlying appointment. If the appointment has already been secured, it will be canceled. If not, the orchestration will not schedule one.

```
DELETE/orchestrations/trade-type/I/unit-nbr/ABCD1234567/locations/D
```

Dual Moves

Dual moves can be orchestrated in two ways:

1. By adding a specific container to the 'dual_unit' section.
2. By adding matching preferences using the 'dual' section under 'reservation_preferences' (Dual Matching). Dual Matching is only available if this service is specifically enabled for the orchestration client user. Otherwise, any values under 'dual' section will be ignored and no dual matching will be performed.

Orchestrating a Dual Move by using the 'dual_unit' section

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "guarantee_through_date": "2022-01-01",
          "reservation_details": {
            "dual_unit": {
              "unit_info": {
                "unit_nbr": "DCBA7654321",
                "unit_category": "C"
              }
            }
          }
        }
      ]
    }
  ]
}
```

Orchestrating a Dual Move by using the 'dual' section under 'reservation_preferences' (Dual Matching)

The `dual` section can be passed to indicate the orchestration to find a container matching the preferences to perform a dual move. Duals can be requested upfront when creating the orchestration, or an orchestration can be modified to request a dual. See examples below.

Create Import Orchestration for the first available reservation with a dual match request on a full container

The following request will create an orchestration to secure an appointment at the first available time slot. If the container is full, match = true will trigger a search for another orchestration with an empty return to the same port. "match_cutoff=60" indicates that the empty container move can be added to the pick move until 60 minutes before the pick import move start. Past that time,

the search for the empty return will stop and the dual move will not be achieved.

Also, threshold configuration for the appointments will be considered when calculating buffer time to select a time slot (see “threshold”). Although “match_cutoff” may be provided to enforce a buffer time, “threshold” configuration on the potential match may render it not eligible if the threshold provided is bigger than the cutoff time.

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "guarantee_through_date": "2022-01-01",
          "reservation_details": {
            "reservation_preferences": {
              "dual": {
                "match": true,
                "match_cutoff": 60,
              }
            }
          }
        }
      ]
    }
  ]
}
```

Create Import Orchestration for a reservation after December 23rd, 2021, anytime from 8:00 am to 12:00 pm, priority for Wednesday, Monday, Thursday, Friday, Tuesday in that order, with a dual match request on an empty container

The following request will create an orchestration to secure an a time slot given the preferences. If the container is empty, match = true will trigger a search for another orchestration with a full container to same port.

If a match is found (active orchestration with a compatible full container flagged with match=true and compatible reservation preferences at the same port), match preferences will apply depending on:

- Match until “2021-12-29”: only an Actual Dual will be matched since VD (Virtual Dual) and SA (Stand Alone) are disabled (VD1=0, SA1=0) in the primary window.
- Match after “2021-12-29” and until “2022-01-01”: Actual Dual is preferred over Virtual Dual. Stand Alone is allowed if no dual could be matched (AD=1, VD=3, SA=3).

Actual Dual: same terminal

Virtual Dual: same port, different terminal OR same terminal at a different time slot

Stand Alone: secure appointment for empty return without a match with a full

POST /orchestrations

```
{
  "order": [
    {
      "trade_type": "I",
      "billoflading": "BOL12345",
      "unit": {
        "unit_nbr": "ABCD1234567"
      },
      "locations": [
        {
          "port_cd": "PORTA",
          "guarantee_through_date": "2022-01-01",
          "reservation_details": {
            "free_time_expiration": "2022-01-01",
            "reservation_preferences": {
              "preference_datetimes": [
                {
                  "range": ["08:00:00", "12:00:00"],
                  "days": [3, 1, 4, 5, 2]
                  "after": "2021-12-23"
                }
              ]
            },
            "dual": {
              "match": true,
              "primary_match_cutoff": "2021-12-29",
              "match_preferences": "SA1=0, VD1=0, AD1=1, SA2=3, AD2=1, VD2=2"
            }
          }
        }
      ]
    }
  ]
}
```

Modify an Import Orchestration with a dual match request on a full container

The following request will modify an existing orchestration on a full container to search for an empty return to create a dual.

PATCH /orchestrations/trade-type/I/unit-nbr/ABCD1234567/locations/O

```
{
  "reservation_preferences": {
    "dual": {
      "match": true,
      "match_cutoff": 60
    }
  }
}
```

Modify an Import Orchestration with a dual match request on an empty container

The following request will modify an existing orchestration on an empty container to search for a full container to create a dual.

PATCH /orchestrations/trade-type/I/unit-nbr/ABCD1234567/locations/D

```
{
  "reservation_preferences": {
    "dual": {
      "match": true,
      "primary_match_cutoff": "2021-12-29",
      "match_preferences": "SA1=0, VD1=0, AD1=1, SA2=3, AD2=1, VD2=2"
    }
  }
}
```

Adding an Empty Return to an existing orchestration and create a Dual Move with another orchestration

Empty Returns can be linked to an existing orchestration to Create a Dual Move (Drop + Pick) if the boxes are compatible with the chassis.

The orchestration of unit DCBA7654321 may or may not have a confirmed appointment. The Pick Import will have priority if time preferences for both appointments are passed. Once the Pick Import is scheduled, the Drop move will be added to that visit.

PATCH /orchestrations/trade-type/I/unit-nbr/ABCD1234567/locations/D

```
{
  "port_cd": "PORTA",
  "reservation_details": {
    "dual_unit": {
      "unit_info": {
        "unit_nbr": "DCBA7654321",
        "unit_category": "C"
      }
    }
  }
}
```

Terminal payments or appointments not supported

Immediately after the orchestration is initiated, the terminal is not known. The terminal will be reported by the system in the container status response (see sections “Container Status” and

“Receiving Notifications”). If the terminal does not support any of the requested services, the following warnings will be triggered.

- 10022 “Terminal not supported for payments” warning if payment guarantee through date was provided and no support for payments.
- 10028 “Terminal not supported for payments. Only third party will be processed” if only third party such as PierPass and CleanTruck are opted in and available.
- 10024 “Terminal not supported for appointments” warning if appointment preferences were provided.

For orchestration updates requests (see “Updating an Orchestration” section), it is possible that by the time of the request, the terminal is already reported. In that case, the request will fail with the following errors:

- 10023 “Terminal not supported for payments” if payment guarantee through date was provided.
- 10025 “Terminal not supported for appointments” if appointment preferences were provided.

Orchestration Status

Request

GET/orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/orchestration-status

Description:

Provides details if orchestration is active or not, if orchestration has any warnings or errors, such as Bill of Lading # not matching, Guarantee Through Date not provided, etc.

Header

See Authentication

Body

None

Responses

200 OK

```
{
  "status": "string",
  "submitted_dttm": "string",
  "initiated_dttm": "string",
  "ended_dttm": "string",
  "messages": [
    {
      "message_cd": "string",
      "message_txt": "string",
      "message_severity_cd": "string"
    }
  ]
}
```

```
}
```

Name	Description
status	<i>string</i> representing the status of the orchestration. ACTIVE, STOPPED.
submitted_dttm	
initiated_dttm	
ended_dttm	
messages	<i>list of object</i> Messages associated to the container orchestration
messages. message_cd	<i>string</i> Message code. Refer to “Error Codes and Severity Levels” for more information.
messages. message_txt	<i>string</i> Message description associated to the orchestration. Refer to “Error Codes and Severity Levels” for more information.
messages. message_severity_cd	<i>string</i> Message severity code associated to the container orchestration. I = Informational W = Warning C = Critical

Container Status

The Container Status Update, outbound messaging, enables customer(s) and stakeholder(s) systems to be automatically updated in near real time as events and availability information change throughout the orchestration lifecycle.

Request

GET /orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/unit-status

Description:

Provides detailed status on a given container’s physical location – i.e. Manifested, On Vessel, In Yard, Gate Out

Header

See Authentication

Body

None

Responses

200 OK

```
{  
  "unitinfo": {  
    "unit_cat": "CONTAINER",  
    "unit_nbr": "ABCD293524",
```

```

"unittypeiso_cd": "22G1",
"unitsztype_cd": "20SD",
"ownerline_scac": "ABCD",
"ownerline_cd": "AB",
"tare_wgt": "2200.0",
"tarewt_unit": "Kilograms",
"lastpretrip_dttm": null
},
"unitstatusinfo": {
  "status_cd": "Y",
  "status_desc": "IN YARD",
  "unituse_cd": "I",
  "unituse_desc": "IMPORT"
},
"locations": [
  {
    "port_cd": "PORTA",
    "port_nm": "Port of A",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A",
    "locationtype_cd": "O",
    "guarantee_through_date": "2022-10-25",
    "reservation_preferences": {
      "try_later_dates": "Y",
      "preference_datetimes": []
    },
  },
  "seq_nbr": "1",
  "locationstatus_cd": "ACT",
  "locationinfo": {
    "arrivalinfo": {
      "carriertype_cd": "VC",
      "carriertype_desc": "VESSEL",
      "vesselinfo": {
        "vessel_cd": "12345678",
        "voyage_nbr": "011A",
        "vesseltype_cd": "L",
        "estarrival_dt": "",
        "actarrival_dt": ""
      }
    }
  },
  "departureinfo": {
    "carriertype_cd": "MC",
    "carriertype_desc": "Motor Carrier",
    "appointment_info": {
      "reservation_status": "CONFIRMED",
      "dual_status": null,
      "ticket_nbr": "ABC-0001-00001",
      "gate_cd": "1234",
      "trucker_info": {
        "trucker_scac": "ABCT",
        "rfid_nbr": "",
        "plate_nbr": "",
        "company_nm": "ABC TRANSPORT INC

```

```

    },
    "driverinfo": {
      "first_nm": "J",
      "last_nm": "Doe"
    },
    "slot_info": {
      "visit_dt": "2022-10-21",
      "visitstart_tm": "07:00",
      "visitend_tm": "07:30",
    },
    "units": [
      {
        "move_type_cd": "IP",
        "unit_nbr": " ABCD1234567",
        "unit_category": "CN",
        "line_scac": "ABCL",
        "sztype_iso": "22G1",
        "sztype_cd": "20SD",
        "yard_spot": "AB10",
        "seal_nbrs": [],
        "spcl_instructions": [
          {
            "instructions": ""
          }
        ]
      }
    ]
  },
  "cargoinfo": {
    "bol_nbr": "ABC12345",
    "reefer_flg": null,
    "temp_min": null,
    "mintemp_uom": null,
    "temp_max": null,
    "maxtemp_uom": null,
    "hazmat_flg": null,
    "imoclass_cd": null,
    "od_flg": null,
    "seal_nbr": null,
    "bcocompany_nm": null,
    "userline_scac": null
  },
  "currentconditioninfo": {
    "damaged_flg": null,
    "yard_loc": "AB10",
    "block_nbr": null,
    "supply_temp": null,
    "return_temp": null,
    "set_temp": null,
    "fullempy_cd": "FULL",
    "gross_wgt": 30480,
  }
}

```

```

        "grosswgt_unit": "Kilograms",
        "actual_wgt": null,
        "actualwgt_unit": null
    }
}
],
"holdinfo": [
    {
        "type": "FEE",
        "status": "RELEASED",
        "status_dttm": "2022-10-20T17:17:34.294476",
        "details": [
            {
                "hold_dttm": "2022-10-20T15:15:51.287916",
                "holdreason_cd": "DWELL",
                "holdreason_desc": "DWELL",
                "release_dttm": "2022-10-20T17:17:34.294476",
                "reportedby": "ABCL"
            }
        ]
    }
],
"fee_receipts": [
    {
        "fee_cd": "DWT",
        "fee_desc": "Extended Dwell Time Fee Through 2022-10-20",
        "fee_amt": "70.00",
        "fee_paid_dttm": "2022-10-20T17:17:34.294476"
    },
    {
        "fee_cd": "4I",
        "fee_desc": "Demurrage Through 2022-10-20",
        "fee_amt": "285.00",
        "fee_paid_dttm": "2022-10-20T17:17:34.294476"
    }
],
"lfdinfo": {
    "good_through_date": "2022-10-20",
    "last_free_day": "2022-10-17"
},
"activity": [
    {
        "seq_nbr": 1,
        "event_cd": "CNMAN",
        "event_desc": "Container Manifested",
        "event_dttm": null,
        "facility_cd": null,
        "facility_nm": null,
    },
    {
        "seq_nbr": 2,

```

```

    "event_cd": "DISCH",
    "event_desc": "Discharged",
    "event_dttm": "2022-10-11T15:13:11",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 3,
    "event_cd": "LFDAY",
    "event_desc": "Last Free Day",
    "event_dttm": "2022-10-16T23:59:00",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 4,
    "event_cd": "PLAVI",
    "event_desc": "Ready for pick up",
    "event_dttm": "2022-10-11T15:16:07",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 5,
    "event_cd": "APNMT",
    "event_desc": "Pregate",
    "event_dttm": "2022-10-20T10:00:00",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 6,
    "event_cd": "DPTRM",
    "event_desc": "Departed Terminal",
    "event_dttm": null,
    "facility_cd": null,
    "facility_nm": null,
  },
  {
    "seq_nbr": 9,
    "event_cd": "ARCUS",
    "event_desc": "Arrived at customer",
    "event_dttm": null,
    "facility_cd": null,
    "facility_nm": null,
  },
  {

```

```

    "seq_nbr": 10,
    "event_cd": "EMTOT",
    "event_desc": "Empty released by customer",
    "event_dttm": null,
    "facility_cd": null,
    "facility_nm": null,
  },
  {
    "seq_nbr": 11,
    "event_cd": "EMTRC",
    "event_desc": "Empty Received",
    "event_dttm": null,
    "facility_cd": null,
    "facility_nm": null,
    "unitevent_uid": ""
  }
]
}

```

Container Status object

Container status fields

Name	Description
unitinfo	<i>object</i> unit information
unitinfo.	<i>string</i>
unit_cat	Unit category
unitinfo.	<i>string</i>
unit_nbr	Unit number
unitinfo.	<i>string</i>
unittypeiso_cd	Unit Size Type ISO Code
unitinfo.	<i>string</i>
unitsztype_cd	eModal Unit Size Type
unitinfo.	<i>string</i>
ownerline_scac	Line SCAC
unitinfo.	<i>string</i>
ownerline_cd	eModal Line Code
unitinfo.	<i>string</i>
tare_wgt	Weight of empty unit

unitinfo. tarewgt_unit	<i>string</i> Tare Weight Unit (i.e. kilograms)
unitinfo. lastpretrip_dttm	
unitstatusinfo	<i>object</i> Unit status information
unitstatusinfo. status_cd	<i>string</i> Status Code (see Container Status Codes)
unitstatusinfo. status_desc	<i>string</i> Status description (see Container Status Codes)
unitstatusinfo. unituse_cd	<i>string</i> Unit Trade Type code I = Import X = Export
unitstatusinfo. unituse_desc	<i>string</i> Unit Trade Type Description Import Export
locations	<i>list of object</i> List of location objects in sequence (by seq_number) visited by the container.
locations. port_cd	<i>string</i> Port Code. Refer to eModal Port Codes for more information on Ports and Terminals
locations. port_nm	<i>string</i> Port Name
locations. facility_cd	<i>string</i> EModal Facility Code
locations. facility_nm	<i>string</i>

	Facility Name
locations. locationtype_cd	<i>string</i> Location type code: O = Origin M = Intermediate (i. e. Warehouse) D = Destination (usually back to the port of origin)
locations. guarantee_through_date	<i>string</i> Guarantee Through Date for fee payments if provided formatted as 'YYYY-MM-DD'
locations. reservation_preferences	<i>object</i> Reservation Preferences originally passed by the user. See create orchestration request.
locations. seq_nbr	<i>string</i> Sequence number of the location in the unit lifecycle.
locations. locationstatus_cd	<i>string</i> Status of the location in the lifecycle. CMP = Completed ACT = Active PND = Pending If multiple locations are represented, only one is flagged as ACT. All previous locations are CMP and future ones are PND. A location is turned active when the system receives a container status from that location. In this way, for a container that's gated out, the active location is still the facility that reported the GATE OUT event. All future locations are PND even if there is an appointment scheduled in one of them (see Container Status for Empty Returns).
locations. locationinfo	<i>object</i>

	Location information
locations. arrivalinfo	<i>object</i> Arrival information
locations.arrivalinfo. carriertype_cd	<i>string</i> Carrier type code MC=Motor Carrier VC=Vessel RC=Rail Carrier
locations.arrivalinfo. carriertype_desc	<i>string</i> Carrier Type description
locations.arrivalinfo. vesselinfo	<i>object</i> Vessel information
locations.arrivalinfo. vesselinfo.vessel_cd	<i>string</i> Vessel code
locations.arrivalinfo. vesselinfo.voyage_nbr	<i>string</i> Voyage Number
locations.arrivalinfo. vesselinfo.vesseltype_cd	<i>string</i> Vessel Type Code
locations.arrivalinfo. vesselinfo.estarrival_dt	<i>string</i> Vessel estimated arrival. YYYY-MM-DD
locations.arrivalinfo.vesselinfo. actarrival_dt	<i>string</i> Vessel actual arrival. YYYY-MM-DD
locations.arrivalinfo. railinfo	<i>object</i> Rail information
locations.arrivalinfo. appointmentinfo	<i>object</i> Appointment information.
locations.arrivalinfo.appointmentinfo. reservation_status	<i>string</i> Reservation Status. See Reservation Status table for reference.
locations.arrivalinfo.appointmentinfo. ticket_nbr	<i>string</i>

	Reservation ticket number.
locations.arrivalinfo.appointmentinfo. gate_cd	<i>string</i> Code required by the terminal to enter gate if applicable. If terminal does not require a code, this field is empty.
locations.arrivalinfo.appointmentinfo. trucker_info	<i>object</i> Trucker information
locations.arrivalinfo.appointmentinfo.trucker_info. trucker_scac	<i>string</i> Trucker SCAC
locations.arrivalinfo.appointmentinfo.trucker_info. rfid_nbr	<i>string</i> RFID number
locations.arrivalinfo.appointmentinfo.trucker_info. plate_nbr	<i>string</i> Plate number
locations.arrivalinfo.appointmentinfo.trucker_info. company_nm	<i>string</i> Company Name
locations.arrivalinfo.appointmentinfo. driverinfo	<i>object</i> Driver information
locations.arrivalinfo.appointmentinfo.driverinfo. first_nm	<i>string</i> Driver's first name
locations.arrivalinfo.appointmentinfo.driverinfo. last_nm	Driver's last name
locations.arrivalinfo.appointmentinfo. slot_info	<i>object</i> Reservation slot information
locations.arrivalinfo.appointmentinfo.slot_info. visit_dt	<i>string</i> Reservation date YYYY-MM-DD (local time)
locations.arrivalinfo.appointmentinfo.slot_info. visitstart_tm	<i>string</i> Slot start time HH:MM (local time)
locations.arrivalinfo.appointmentinfo.slot_info. visitend_tm	<i>string</i> Slot end time HH:MM (local time)

locations.arrivalinfo.appointmentinfo. units	<i>list of object</i> List of additional units on the same reservation if applies. i. e. Dual or Twin move.
locations.arrivalinfo.appointmentinfo.units. move	<i>string</i> Move Type of the additional unit P=Pick D=Drop
locations.arrivalinfo.appointmentinfo.units. unit_nbr	
locations.arrivalinfo.appointmentinfo.units. unit_category	
locations.arrivalinfo.appointmentinfo.units. line_scac	
locations.arrivalinfo.appointmentinfo.units. sztype_iso	
locations.arrivalinfo.appointmentinfo.units. sztype_cd	
locations.arrivalinfo.appointmentinfo.units. yard_spot	<i>string</i> Additional information about location within yard
locations.arrivalinfo.appointmentinfo.units. seal_nbrs	<i>list of string</i> Seal Numbers
locations.arrivalinfo.appointmentinfo.units. spcl_instructions	<i>list of string</i> Instructions
locations.arrivalinfo.appointmentinfo.units.spcl_instructions. instruction	<i>string</i> Special instruction
locations. departureinfo	<i>object</i> Same as arrival info

Orchestration Status for Empty Returns

If the container has gated out from the facility, the orchestration is now ready to receive a drop empty (Empty Return) appointment request (see “Empty Returns” section above). Once the request is accepted, another location will appear in the ‘locations’ array with ‘locationstatus_cd’ = ‘PND’ (Pending). This location will hold the information regarding the requested appointment. Please note that the active location continues to be the last facility to report the container.

See example below. Locations array has two locations: 'locationtype_cd' = 'D' (destination) holds the information about the empty return appointment. Appointment information is located under "arrivalinfo":

```
{
  "unitinfo": {
    "unit_cat": "CONTAINER",
    "unit_nbr": "ABCD1234567",
    "unittypeiso_cd": "22G1",
    "unitsztype_cd": "20SD",
    "ownerline_scac": "ABCD",
    "ownerline_cd": "AB",
    "tare_wgt": "2200.0",
    "tarewt_unit": "Kilograms",
    "lastpretrip_dttm": null
  },
  "unitstatusinfo": {
    "status_cd": "O"
    "status_desc": "GATE OUT",
    "unituse_cd": "I",
    "unituse_desc": "IMPORT"
  },
  "locations": [
    {
      "port_cd": "PORTA",
      "port_nm": "Port of A",
      "facility_cd": "FCLTA",
      "facility_nm": "Facility A",
      "locationtype_cd": "O",
      "guarantee_through_date": "2022-10-25",
      "reservation_preferences": null,
      "seq_nbr": "1",
      "locationstatus_cd": "ACT",
      "locationinfo": {
        "arrivalinfo": {
          "carriertype_cd": "VC",
          "carriertype_desc": "VESSEL",
          "vesselinfo": {
            "vessel_cd": "12345678",
            "voyage_nbr": "011A",
            "vesseltype_cd": "L",
            "estarrival_dt": "",
            "actarrival_dt": ""
          }
        }
      },
      "departureinfo": {
        "carriertype_cd": "MC",
        "carriertype_desc": "Motor Carrier",
        "appointment_info": {
          "reservation_status": "COMPLETED",

```

```

    "ticket_nbr": "FCLTA-1234-1234",
    "gate_cd": "1234",
    "trucker_info": {
      "trucker_scac": "ABCT",
      "rfid_nbr": "",
      "plate_nbr": "",
      "company_nm": "ABCT TRANSPORT INC"
    },
    "driverinfo": {
      "first_nm": "J",
      "last_nm": "Doe"
    },
    "slot_info": {
      "visit_dt": "2022-10-21",
      "visitstart_tm": "07:00",
      "visitend_tm": "07:30",
    },
    "units": [
      {
        "move": "IP",
        "unit_nbr": " ABCD1234567",
        "unit_category": "CN",
        "line_scac": "ABCL",
        "sztype_iso": "22G1",
        "sztype_cd": "20SD",
        "yard_spot": "B013",
        "seal_nbrs": [],
        "spcl_instructions": [
          {
            "instructions": ""
          }
        ]
      }
    ]
  },
  "cargoinfo": {
    "bol_nbr": "ABC12345",
    "reefer_flg": null,
    "temp_min": null,
    "mintemp_uom": null,
    "temp_max": null,
    "maxtemp_uom": null,
    "hazmat_flg": null,
    "imoclass_cd": null,
    "od_flg": null,
    "seal_nbr": null,
    "bcocompany_nm": null,
    "userline_scac": null
  },
  "currentconditioninfo": {
    "damaged_flg": null,
    "yard_loc": "AB10",

```

```

        "block_nbr": null,
        "supply_temp": null,
        "return_temp": null,
        "set_temp": null,
        "fullempy_cd": "FULL",
        "gross_wgt": 30480,
        "grosswgt_unit": "Kilograms",
        "actual_wgt": null,
        "actualwgt_unit": null
    }
},
{
    "port_cd": "PORTA",
    "port_nm": "Port of A",
    "facility_cd": "FCTLA",
    "facility_nm": "Facility A",
    "locationtype_cd": "D",
    "guarantee_through_date": null,
    "reservation_preferences": null,
    "seq_nbr": "2",
    "locationstatus_cd": "PND",
    "locationinfo": {
        "arrivalinfo": {
            "carriertype_cd": "MC",
            "carriertype_desc": "Motor Carrier",
            "vesselinfo": null,
            "carrier_cd": null,
            "appointment_info": {
                "reservation_status": "CONFIRMED",
                "ticket_nbr": "FCTA-4321-4321",
                "gate_code": "",
                "trucker_info": {
                    "trucker_scac": "ABCT",
                    "rfid_nbr": "",
                    "plate_nbr": "",
                    "company_nm": null
                },
                "driverinfo": {
                    "first_nm": "",
                    "last_nm": ""
                },
                "slot_info": {
                    "visit_dt": "2022-09-29",
                    "visitstart_tm": "08:00",
                    "visitend_tm": "09:00"
                },
                "units": [
                    {
                        "move": "MD",
                        "unit_nbr": " ABCD1234567",
                        "unit_category": "CN",
                        "line_scac": "ABCL",

```

```

        "sztype_iso": "4270",
        "sztype_cd": null,
        "yard_spot": "",
        "seal_nbrs": [],
        "spcl_instructions": [
            {
                "instructions": ""
            }
        ]
    }
},
"departureinfo": null,
"cargoinfo": {
    "bol_nbr": "ABC12345",
    "reefer_flg": null,
    "temp_min": null,
    "mintemp_uom": null,
    "temp_max": null,
    "maxtemp_uom": null,
    "hazmat_flg": null,
    "imoclass_cd": null,
    "od_flg": null,
    "seal_nbr": null,
    "bcocompany_nm": null,
    "userline_scac": null
},
"currentconditioninfo": {
    "damaged_flg": null,
    "yard_loc": "AB10",
    "block_nbr": null,
    "supply_temp": null,
    "return_temp": null,
    "set_temp": null,
    "fullempy_cd": "FULL",
    "gross_wgt": 30480,
    "grosswgt_unit": "Kilograms",
    "actual_wgt": null,
    "actualwgt_unit": null
}
}
},
"holdinfo": [
    {
        "type": "FEE",
        "status": "RELEASED",
        "status_dttm": "2022-10-20T17:17:34.294476",
        "details": [
            {

```

```

        "hold_dttm": "2022-10-20T15:15:51.287916",
        "holdreason_cd": "DWELL",
        "holdreason_desc": "DWELL",
        "release_dttm": "2022-10-20T17:17:34.294476",
        "reportedby": "ABCL"
    }
  ]
},
"fee_receipts": [
  {
    "fee_cd": "DWT",
    "fee_desc": "Extended Dwell Time Fee Through 2022-10-20",
    "fee_amt": "70.00",
    "fee_paid_dttm": "2022-10-20T17:17:34.294476"
  },
  {
    "fee_cd": "4I",
    "fee_desc": "Demurrage Through 2022-10-20",
    "fee_amt": "285.00",
    "fee_paid_dttm": "2022-10-20T17:17:34.294476"
  }
],
"lfdinfo": {
  "good_through_date": "2022-10-20",
  "last_free_day": "2022-10-17"
},
"activity": [
  {
    "seq_nbr": 1,
    "event_cd": "CNMAN",
    "event_desc": "Container Manifested",
    "event_dttm": null,
    "facility_cd": null,
    "facility_nm": null,
  },
  {
    "seq_nbr": 2,
    "event_cd": "DISCH",
    "event_desc": "Discharged",
    "event_dttm": "2022-10-11T15:13:11",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A",
  },
  {
    "seq_nbr": 3,
    "event_cd": "LFDAY",
    "event_desc": "Last Free Day",
    "event_dttm": "2022-10-17T23:59:00",
    "facility_cd": "FCLTA",
  }
]

```

```

    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 4,
    "event_cd": "PLAVI",
    "event_desc": "Ready for pick up",
    "event_dttm": "2022-10-11T15:16:07",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 5,
    "event_cd": "APNMT",
    "event_desc": "Pregate",
    "event_dttm": "2022-10-21T07:00:00",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 6,
    "event_cd": "DPTRM",
    "event_desc": "Departed Terminal",
    "event_dttm": "2022-10-21T07:30:00",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 9,
    "event_cd": "ARCUS",
    "event_desc": "Arrived at customer",
    "event_dttm": null,
    "facility_cd": null,
    "facility_nm": null,
  },
  {
    "seq_nbr": 10,
    "event_cd": "EMTOT",
    "event_desc": "Empty released by customer",
    "event_dttm": null,
    "facility_cd": null,
    "facility_nm": null,
  },
  {
    "seq_nbr": 11,
    "event_cd": "EMTRC",
    "event_desc": "Empty Received",

```

```

    "event_dttm": null,
    "facility_cd": null,
    "facility_nm": null,
    "unitevent_uid": ""
  }
]
}

```

Orchestration Status for Off-Dock Locations

If the container is moved to an off-dock location, a new location will appear in the 'locations' array with 'locationtype_cd' = 'M' (Intermediate). The origin facility will report a GATE OUT status and will be marked as completed ('locationstatus_cd' = 'CMP') and the off-dock location will report a GATE IN status and will become active ('locationstatus_cd' = 'ACT').

```

{
  ...
  "unitstatusinfo": {
    "status_cd": "O",
    "status_desc": "GATE IN",
    "unituse_cd": "I",
    "unituse_desc": "IMPORT"
  },
  "locations": [
    {
      "port_cd": "PORTA",
      "port_nm": "Port of A",
      "facility_cd": "FCLTA",
      "facility_nm": "Facility A",
      "locationtype_cd": "O",
      "guarantee_through_date": "2022-10-25",
      "reservation_preferences": null,
      "seq_nbr": "1",
      "locationstatus_cd": "CMP",
      "locationinfo": {
        ...
        "departureinfo": {
          "carrier_cd": "MC",
          "carrier_desc": "Motor Carrier",
          ...
        },
        ...
      },
      ...
    }
  ],
  {
    "port_cd": "PORTA",
    "port_nm": "Port of A",
    "facility_cd": "OFFDCKA",
    "facility_nm": "Off-Dock Facility A",
  }
}

```

```

"locationtype_cd": "M",
"guarantee_through_date": null,
"reservation_preferences": null,
"seq_nbr": "2",
"locationstatus_cd": "ACT",
"locationinfo": {
  "arrivalinfo": {
    ...
  },
  "departureinfo": {
    ...
  },
  ...
}
},
"holdinfo": [
  {
    "type": "FEE",
    "status": "RELEASED",
    "status_dttm": "2022-10-20T17:17:34.294476",
    "details": [
      {
        "hold_dttm": "2022-10-20T15:15:51.287916",
        "holdreason_cd": "DWELL",
        "holdreason_desc": "DWELL",
        "release_dttm": "2022-10-20T17:17:34.294476",
        "reportedby": "ABCL"
      }
    ]
  }
],
"fee_receipts": [
  {
    "fee_cd": "DWT",
    "fee_desc": "Extended Dwell Time Fee Through 2022-10-20",
    "fee_amt": "70.00",
    "fee_paid_dttm": "2022-10-20T17:17:34.294476"
  },
  {
    "fee_cd": "4I",
    "fee_desc": "Demurrage Through 2022-10-20",
    "fee_amt": "285.00",
    "fee_paid_dttm": "2022-10-20T17:17:34.294476"
  }
],
"lfdinfo": {
  "good_through_date": "2022-10-20",
  "last_free_day": "2022-10-17"
},
"activity": [
  {

```

```

    "seq_nbr": 1,
    "event_cd": "CNMAN",
    "event_desc": "Container Manifested",
    "event_dttm": null,
    "facility_cd": null,
    "facility_nm": null,
  },
  {
    "seq_nbr": 2,
    "event_cd": "DISCH",
    "event_desc": "Discharged",
    "event_dttm": "2022-10-11T15:13:11",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A",
  },
  {
    "seq_nbr": 3,
    "event_cd": "LFDAY",
    "event_desc": "Last Free Day",
    "event_dttm": "2022-10-17T23:59:00",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 4,
    "event_cd": "PLAVI",
    "event_desc": "Ready for pick up",
    "event_dttm": "2022-10-11T15:16:07",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 5,
    "event_cd": "APNMT",
    "event_desc": "Pregate",
    "event_dttm": "2022-10-21T07:00:00",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 6,
    "event_cd": "DPTRM",
    "event_desc": "Departed Terminal",
    "event_dttm": "2022-10-21T07:30:00",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  }

```

```

    },
    {
      "seq_nbr": 9,
      "event_cd": "ARCUS",
      "event_desc": "Arrived at customer",
      "event_dttm": null,
      "facility_cd": null,
      "facility_nm": null,
    },
    {
      "seq_nbr": 10,
      "event_cd": "EMTOT",
      "event_desc": "Empty released by customer",
      "event_dttm": null,
      "facility_cd": null,
      "facility_nm": null,
    },
    {
      "seq_nbr": 11,
      "event_cd": "EMTRC",
      "event_desc": "Empty Received",
      "event_dttm": null,
      "facility_cd": null,
      "facility_nm": null,
      "unitevent_uid": ""
    }
  ]
}

```

Orchestration Status for Dual Matching

For containers that are part of the dual matching, the new “dual_status” section will be sent as part of the “appointment_info” attribute, indicating the “matched”, “pending match” or “no match” state.

Once a match has been established, the “units” entry for the dual container will indicate a “dual_type” of either "AD" (Actual Dual) or "VD" (Virtual Dual).

```

{
  "unitinfo": {
    "unit_cat": "CONTAINER",
    "unit_nbr": "ABCD1234567",
    ...
  },
  ...
  "locations": [
    {

```

```

"port_cd": "PORTA",
"port_nm": "Port of A",
"facility_cd": "FCTLA",
"facility_nm": "Facility A",
"locationtype_cd": "0",
"guarantee_through_date": null,
"reservation_preferences": null,
"seq_nbr": "1",
"locationstatus_cd": "ACT",
"locationinfo": {
  "arrivalinfo": {
    "carriertype_cd": "MC",
    "carriertype_desc": "Motor Carrier",
    "vesselinfo": null,
    "carrier_cd": null,
    "appointment_info": {
      "reservation_status": "CONFIRMED",
      "dual_status": "MATCHED",
      "ticket_nbr": "FCTA-4321-4321",
      "gate_code": "",
      "trucker_info": {
        "trucker_scac": "ABCT",
        "rfid_nbr": "",
        "plate_nbr": "",
        "company_nm": null
      },
      "driverinfo": {
        "first_nm": "",
        "last_nm": ""
      },
      "slot_info": {
        "visit_dt": "2022-09-29",
        "visitstart_tm": "08:00",
        "visitend_tm": "09:00"
      },
      "units": [
        {
          "move": "IP",
          "unit_nbr": "ABCD1234567",
          "unit_category": "CN",
          "line_scac": "ABCL",
          "sztype_iso": "4270",
          ...
        },
        {
          "move": "MD",
          "dual_type": "AD",
          "unit_nbr": "BCDA1234567",
          "unit_category": "CN",
          "line_scac": "ABCL",
          "sztype_iso": "4270",
          ...
        }
      ]
    }
  }
}

```

```

    },
    ...
  }
}
],
...
',
"activity": [
  {
    "seq_nbr": 1,
    "event_cd": "CNMAN",
    "event_desc": "Container Manifested",
    "event_dttm": null,
    "facility_cd": null,
    "facility_nm": null,
  },
  {
    "seq_nbr": 2,
    "event_cd": "DISCH",
    "event_desc": "Discharged",
    "event_dttm": "2022-10-11T15:13:11",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A",
  },
  {
    "seq_nbr": 3,
    "event_cd": "LFDAY",
    "event_desc": "Last Free Day",
    "event_dttm": "2022-10-17T23:59:00",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 4,
    "event_cd": "PLAVI",
    "event_desc": "Ready for pick up",
    "event_dttm": "2022-10-11T15:16:07",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
  {
    "seq_nbr": 5,
    "event_cd": "APNMT",
    "event_desc": "Pregate",
    "event_dttm": "2022-10-21T07:00:00",
    "facility_cd": "FCLTA",
    "facility_nm": "Facility A"
  },
},

```

```
{
  "seq_nbr": 6,
  "event_cd": "DPTRM",
  "event_desc": "Departed Terminal",
  "event_dttm": "2022-10-21T07:30:00",
  "facility_cd": "FCLTA",
  "facility_nm": "Facility A"
},
{
  "seq_nbr": 9,
  "event_cd": "ARCUS",
  "event_desc": "Arrived at customer",
  "event_dttm": null,
  "facility_cd": null,
  "facility_nm": null,
},
{
  "seq_nbr": 10,
  "event_cd": "EMTOT",
  "event_desc": "Empty released by customer",
  "event_dttm": null,
  "facility_cd": null,
  "facility_nm": null,
},
{
  "seq_nbr": 11,
  "event_cd": "EMTRC",
  "event_desc": "Empty Received",
  "event_dttm": null,
  "facility_cd": null,
  "facility_nm": null,
  "unitevent_uid": ""
}
]
}
```

Receiving notifications

Status updates can be received via Azure Service Bus. For additional information on how to interact with the Azure Service Bus please refer to the [Service Bus Runtime REST documentation](#).

The following example retrieves a message and deletes it from the ASB queue (destructive read).

Request

DELETE https://{serviceName}.servicebus.windows.net/{provided by Envio 360}/subscriptions/orchestrationupdates/messages/head

Header

Key	Value
Authorization	Provided by Envio 360

Notification messages contain a JSON object with orchestration and container status. See example below:

```
{
  "orchestration_request": {
    "trade_type": "string",
    "unit_nbr": "string",
    "notification_type": [],
    "sections_updated": []
  },
  "orchestration_status": {
    "status": "string",
    "submitted_dttm": "string",
    "initiated_dttm": "string",
    "ended_dttm": "string",
    "messages": [
      {
        "message_cd": "string",
        "message_txt": "string",
        "message_severity_cd": "string"
      }
    ]
  },
  "unit_status": {
    "unitinfo": {
      "unit_cat": "string",
      "unit_nbr": "string",
      "unittypeiso_cd": "string",
      ...
    },
    ...
  },
  ...
}
```

```
}
```

Notification object

Name	Description
orchestration_request	<i>object</i> Information relevant to the notification
orchestration_request.trade_type	<i>string</i> Trade Type
orchestration_request.unit_nbr	<i>string</i> Unit Number.
orchestration_request.notification_type	<i>list of string</i> List of types of notification. “Appointment” indicates a notification triggered by the appointment system. An empty list indicates a general notification.
orchestration_request.sections_updated	<i>list of string</i> List of sections with modifications since the previous notification. Possible values are: “orchestration_status”, “unitinfo”, “unitstatusinfo”, “locations”, “holdinfo”, “fee_receipts”, “lfdinfo” and “activity”. Please refer to “Container Status object” for more information on these sections.
orchestration_status	<i>object</i> Same as orchestration status object
unit_status	<i>object</i> Same as container status object

Notification types

Every notification represents the latest orchestration state (appointments, payments made on behalf of the integrator, container status and error messages and warnings).

Each notification is triggered by internal system notification of changes of any fields of the container data, appointment data, successful payments and when an error/warning occurs.

The usual sequence of messages for an orchestration with no fees on the container and appointments is the following:

1. Orchestration status ACTIVE. Container Status empty. – *orchestration is functional, but the container has not been located yet.*
2. Container Status with data – no appointment has been requested yet.
3. ‘reservation_preferences’ field is populated with selected preferences. – *appointment request is about to be initiated.*
4. Appointment data is set, and status is set to PENDING – *appointment is being requested and pending slot confirmation.*
5. ‘reservation_preferences’ field is cleared. Appointment data is set with ticket number and slot information. – *slot is secured.*

Invalid request

If the body received by the create orchestration endpoint does not receive the expected format (JSON as described in the Create Orchestration section), an HTTP 400 error will be returned, and no orchestration will be created. Calling any of the status endpoints for the container will return HTTP 404 NOT FOUND.

Invalid or missing required field

If the body received by the create orchestration endpoint does not contain all required fields, or any of them does not match the expected format, such as a wrong or missing Port Code, an HTTP 412 error will be returned, and no orchestration will be created. Calling any of the status endpoints for the container will return HTTP 404 NOT FOUND.

Unit not found

On submitting an orchestration, if the port code is valid and the unit number matches the expected format, the orchestration creation request is accepted even if the unit is not yet reported to Envio360 or the provided port does not match.

In this case, a notification will be sent informing that the unit was not found. The orchestration will be stopped automatically.

It may be possible that Envio360 does not receive information about the unit until after it has been unloaded from the vessel for pick imports. The integrator logic should retry creating the orchestration on regular intervals until it is found.

Orchestration created with warnings

If all required parameters are correctly received the Create Orchestration endpoint returns HTTP 202 ACCEPTED. The orchestration is now operating. However, if any of the fields necessary to orchestrate Payments or Appointments are missing, the Orchestration Status endpoint shows all the warnings and errors generated by the orchestration system.

Note: calling the Orchestration Status endpoint always returns HTTP 200 for an existing orchestration -as this indicates a successful query to the system- but the true status of the orchestration should be extracted from the response JSON i. e. missing BOL will block the orchestration until it is indicated that the orchestration is allowed to proceed without a matching BOL by updating the orchestration.

Orchestration stopped

After the orchestration request is successfully accepted, the orchestration can be stopped by any of the following events:

- Container not found
- Container gated in
- Orchestration canceled by integrator

If any of the above occur, the orchestration will be stopped automatically, and a notification will be sent to the integrator. See section “Receiving notifications”.

```
{
  "orchestration_request": {
    "trade_type": "I",
    "unit_nbr": "ABCD1234567",
    "notification_type": [],
  },
  "orchestration_status": {
    "status": "STOPPED",
    ...
  },
  "unit_status": {
    ...
  }
}
```

Non-matching bill of lading

If bill of lading is provided but does not match the one reported by eModal, the Orchestration adds a warning to the messages list:

GET/orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/orchestration-status
200 OK

```
{
  "status": "ACTIVE",
  "submitted_dttm": "2023-01-01T13:36:41.3270000",
  "initiated_dttm": "2023-01-01T13:36:41.6470000",
  "ended_dttm": null,
  "messages": [
    {
      "message_cd": "10015",
      "message_txt": "Non-matching Bill of Lading",
      "message_severity_cd": "W"
    },
    {
      "message_cd": "10007",
      "message_txt": "Missing Guarantee Through Date",
      "message_severity_cd": "W"
    }
  ]
}
```

```
}
]
}
```

Now, orchestration can be updated to indicate that payments are allowed to be resumed despite the BOL discrepancy:

PATCH /orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/locations/{location_type}

```
{
  "guarantee_through_date": "2023-01-05"
}
```

One warning will be removed and the orchestration will resume with paying fees. The BOL warning will remain but will not affect the orchestration in any way.

GET/orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/orchestration-status

200 OK

```
{
  "status": "ACTIVE",
  "submitted_dttm": "2023-01-01T13:36:41.3270000",
  "initiated_dttm": "2023-01-01T13:36:41.6470000"
  "ended_dttm": null,
  "messages": [
    {
      "message_cd": "10015",
      "message_txt": "Non-matching Bill of Lading",
      "message_severity_cd": "W"
    }
  ]
}
```

Third-Party Fees

Only Third-Party Fees supported

If third party fees are supported i.e. PierPASS fees, but no other functionality is available for a terminal, the orchestration submits a claim for those fees and stops.

GET/orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/orchestration-status

404 NOT FOUND

GET /orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/unit-status

404 NOT FOUND

Other Functionality Supported

If third-party fees are supported and other functionality (appointments or terminal payments) is supported as well, the orchestration stays active:

GET/orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/orchestration-status

200 OK

```
{
  "status": "ACTIVE",
  "submitted_dttm": "2023-01-01T13:36:41.3270000",
  "initiated_dttm": "2023-01-01T13:36:41.6470000"
  "ended_dttm": null,
  "messages": []
}
```

The orchestration continues to perform other fee payments and appointment scheduling.

Appointments not supported

If visibility and payments are supported for a terminal, but there is not appointment scheduling available, the orchestration reports accordingly:

GET/orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/orchestration-status
200 OK

```
{
  "status": "ACTIVE",
  "submitted_dttm": "2023-01-01T13:36:41.3270000",
  "initiated_dttm": "2023-01-01T13:36:41.6470000"
  "ended_dttm": null,
  "messages": [
    {
      "message_cd": "10024",
      "message_txt": "Terminal not supported for appointments",
      "message_severity_cd": "W"
    }
  ]
}
```

If an attempt to update appointment details if appointments are not supported:

PATCH /orchestrations/trade-type/l/unit-nbr/ABCD1234567/locations/O

```
{
  "reservation_details": {}
}
```

412 CONFLICT

```
{
  "status": "ACTIVE",
  "submitted_dttm": "2023-01-01T13:36:41.3270000",
  "initiated_dttm": "2023-01-01T13:36:41.6470000"
  "ended_dttm": null,
  "messages": [
    {
      "message_cd": "10025",
      "message_txt": "Terminal not supported for appointments",
      "message_severity_cd": "C"
    }
  ]
}
```

```
}  
]  
}
```

Error code is different than 10023 as the severity is “critical”. The request is rejected.

Appointment confirmed

If preferences match availability and no blockers are detected, an appointment is confirmed. Please note that even with a confirmed appointment, future cancelation events can occur.

Appointment is secured

GET/orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/orchestration-status

200 OK

```
{  
  "status": "ACTIVE",  
  "submitted_dttm": "2023-01-01T13:36:41.3270000",  
  "initiated_dttm": "2023-01-01T13:36:41.6470000"  
  "ended_dttm": null,  
  "messages": []  
}
```

GET /orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/unit-status

200 OK

```
{  
  "event_cat": "CTR",  
  "event_cd": "CNUPD",  
  "unitinfo": {  
    "unit_cat": "CONTAINER",  
    "unit_nbr": "ABCD1234567",  
    ...  
  },  
  "unitstatusinfo": {  
    "status_cd": "Y",  
    "status_desc": "IN YARD",  
    "unituse_cd": "I",  
    "unituse_desc": "IMPORT"  
  },  
  "locations": [  
    {  
      "port_cd": "PORTA",  
      "port_nm": "Port of A",  
      "facility_cd": "FCLTA",  
      "facility_nm": "Facility A",  
      "locationtype_cd": "O",  
      "guarantee_through_date": "2023-01-05",  
      ...  
    },  
    ...  
  ],  
  "locationinfo": {
```

```

...
...
  },
  "departureinfo": {
    "carriertype_cd": "MC",
    "carriertype_desc": "Motor Carrier",
    "appointment_info": {
      "reservation_status": "CONFIRMED",
      "ticket_nbr": "FCLTA -0001-00001",
      "pin_nbr": "1234",
      "trucker_info": {
        "trucker_scac": "ABCT",
        "rfid_nbr": "",
        "plate_nbr": "",
        "company_nm": "ABCD TRANSPORT INC"
      },
      "driverinfo": {
        "first_nm": "J",
        "last_nm": "Doe"
      },
      "slot_info": {
        "visit_dt": "2022-10-25",
        "visitstart_tm": "07:00",
        "visitend_tm": "07:30",
      },
    },
    ...
  },
  ...
  ...
],
"holdinfo": [
  {
    "type": "OTHER",
    "status": "RELEASED",
    "status_dttm": "2023-01-01T17:17:34.294476",
    "details": [
      {
        "hold_dttm": "2023-01-01T15:15:51.287916",
        "holdreason_cd": "CF",
        "holdreason_desc": "CTF Hold",
        "release_dttm": "2023-01-01T17:17:34.294476",
        "reportedby": "FCLTA"
      }
    ]
  },
  {
    "type": "OTHER",
    "status": "RELEASED",
    "status_dttm": "2023-01-01T17:17:34.294476",
    "details": [
      {
        "hold_dttm": "2023-01-01T15:15:51.287916",
        "holdreason_cd": "TM",
        "holdreason_desc": "TMF Hold",
        "release_dttm": "2023-01-01T17:17:34.294476",
        "reportedby": "FCLTA"
      }
    ]
  }
]

```

```

    ]
  }
]
"fee_receipts": [
  {
    "fee_cd": "PIERPASS",
    "fee_desc": "TMF Fee",
    "fee_amt": "68.42",
    "fee_paid_dttm": "2023-01-01T00:00:00.0000000"
  },
  {
    "fee_cd": "PIERPASS",
    "fee_desc": "CTF Fee",
    "fee_amt": "20.00",
    "fee_paid_dttm": "2023-01-01T00:00:00.0000000"
  }
],
...
...
...

```

Appointment confirmed with warnings

A slot may be confirmed by the terminal subject to conditions to be fulfilled prior to the execution. On example of this would be a hold. Some terminals may submit a confirmation with an active fee hold, expecting the company to clear it before executing the move.

Appointment is secured with pending actions

GET/orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/orchestration-status

200 OK

```

{
  "status": "ACTIVE",
  "submitted_dttm": "2023-01-01T13:36:41.3270000",
  "initiated_dttm": "2023-01-01T13:36:41.6470000"
  "ended_dttm": null,
  "messages": []
}

```

GET /orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/unit-status

200 OK

```

{
  "event_cat": "CTR",
  "event_cd": "CNUPD",
  "unitinfo": {...},
  "unitstatusinfo": {...},
  "locations": [
    {
      ...
    }
  ]
  "locationinfo": {

```

```

...
...
,
"departureinfo": {
  ...
  "appointment_info": {
    "reservation_status": "*CONFIRMED",
    "ticket_nbr": " FCLTA -0001-00001",
    "pin_nbr": "1234",
    "trucker_info": {...},
    "driverinfo": {...},
    "slot_info": {...},
    ...
  },
  ...
},
],
"holdinfo": [
  {
    "type": "OTHER",
    "status": "HOLD",
    "status_dttm": "2023-01-01T02:45:13.124999",
    "details": [
      {
        "hold_dttm": "2022-12-30T09:45:06",
        "holdreason_cd": "TM",
        "holdreason_desc": "TMF Hold",
        "release_dttm": null,
        "reportedby": " FCLTA"
      }
    ]
  },
  {
    "type": "OTHER",
    "status": "HOLD",
    "status_dttm": "2023-01-01T02:45:13.124999",
    "details": [
      {
        "hold_dttm": "2022-12-30T09:45:06",
        "holdreason_cd": "CF",
        "holdreason_desc": "CTF Hold",
        "release_dttm": null,
        "reportedby": " FCLTA"
      }
    ]
  }
],
"fee_receipts": [],
...
...
...

```

Appointment canceled by external action

If an appointment has been secured by the orchestration but it is later canceled by external action, such as terminal cancelation or trucker company cancelation via eModal web application, the appointment status will be reported accordingly:

Appointment is canceled

GET/orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/orchestration-status

200 OK

```
{
  "status": "ACTIVE",
  "submitted_dttm": "2023-01-01T13:36:41.3270000",
  "initiated_dttm": "2023-01-01T13:36:41.6470000"
  "ended_dttm": null,
  "messages": []
}
```

GET /orchestrations/trade-type/{trade type}/unit-nbr/{unit_nbr}/unit-status

200 OK

```
{
  "event_cat": "CTR",
  "event_cd": "CNUPD",
  "unitinfo": {
    "unit_cat": "CONTAINER",
    "unit_nbr": "{unit_nbr}",
    ...
  },
  "unitstatusinfo": {
    "status_cd": "Y",
    "status_desc": "IN YARD",
    "unituse_cd": "I",
    "unituse_desc": "IMPORT"
  },
  "locations": [
    {
      "port_cd": "PORTA",
      "port_nm": "Port of A",
      "facility_cd": "FCLTA",
      "facility_nm": "Facility A",
      "locationtype_cd": "O",
      "guarantee_through_date": "2023-01-05",
      ...
    }
  ],
  "locationinfo": {
    ...
  },
  "departureinfo": {
    "carriertype_cd": "MC",
    "carriertype_desc": "Motor Carrier",
    "appointment_info": {
      "reservation_status": "CANCELED",
    }
  }
}
```

```

        "ticket_nbr": "ABC-0001-00001",
        "pin_nbr": "1234",
        "trucker_info": {...},
        "driverinfo": {...},
        "slot_info": {
            "visit_dt": "2022-10-25",
            "visitstart_tm": "07:00",
            "visitend_tm": "07:30",
        },
        ...
    ],
    "holdinfo": [...],
    "fee_receipts": [...]
    ...
    ...
    ...

```

Re-submit request to schedule an appointment on the first available slot

PATCH /orchestrations/trade-type/l/unit-nbr/ABCD1234567/locations/O

```

{
  "reservation_details": {}
}

```

Error codes and severity levels

Code	Message	Severity	Details
10000	Orchestration created successfully	I	Returned by API response on create orchestration.
10001	No orchestration data found	C	Returned by API response on create orchestration if orchestration data is not passed. Orchestration request is rejected.
10002	Invalid trade type	C	Returned by API response on create orchestration if trade type code is not recognized. Orchestration request is rejected.
10003	Invalid port code	C	Returned by API response on create orchestration if port code is not recognized. Orchestration request is rejected.
10004	Container# {container number} is invalid	C	Returned by API response on create orchestration if container number does not match the expected format. Orchestration request is rejected.
10005	Missing Bill of Lading	C	Returned by API response on create orchestration if Bill of Lading is not passed. Orchestration request is rejected. If Bill of

			Lading is passed, but does not match Envio360's records, the Guarantee Through Date will be removed from the orchestration to block accidental payments.
10006	Invalid Guarantee Through Date	C	Returned by API response on create orchestration if Guarantee Through Date does not match the expected format. Orchestration request is rejected.
10007	Missing Guarantee Through Date	W	Returned by API response and notifications. Payment orchestration will not be initiated. Even if passed correctly, this error will appear in notifications if Bill of Lading does not match.
10008	Invalid preferences	C	Returned by API response. Orchestration request is rejected.
10009	Orchestration not found	C	Returned by API on update orchestration if the orchestration does not exist. Request is rejected.
10010	Orchestration canceled successfully	I	Returned by API response on orchestration cancelation request. All orchestration activities will be stopped.
10011	Appointment canceled successfully	I	Returned by API response on appointment cancelation request. Orchestration will continue to report status and process payments if applicable.
10012	{Internal Appointment System message} [Internal Appointment System code]	I	Returned by notifications on events received by the internal appointment system.
10013	{Internal Appointment System message} [Internal Appointment System code]	W	Returned by notifications on events received by the internal appointment system, i.e. no time slots found but orchestration keeps looking, or truck identification required prior to pick up.
10014	{Internal Appointment System message} [Internal Appointment System code]	C	Returned by notifications on events received by the internal appointment system, i.e. no time slots found and threshold exceeded all the options. This error requires action as the appointment attempt has failed and will not be retried.
10015	Non-matching Bill of Lading	W	Returned by notifications. This message is installed if passed BoL does not match Envio 360's records. Also, if Envio 360's has no record of the BoL.
10016	Missing container number	C	Returned by API response on orchestration creation request. Request is rejected.
10017	Missing trade type	C	Returned by API response on orchestration creation request. Request is rejected.
10018	Missing locations list	C	Returned by API response on orchestration creation request. Request is rejected.
10019	Missing port code	C	Returned by API response on orchestration creation request. Request is rejected.
10020	An orchestration is already active for this container	C	Returned by API response on orchestration creation request if the container is already orchestrated by another party. Request is rejected.

10021	Container not found.	C	Returned by API response on container status request.
10022	Terminal not supported for payments	W	Returned by notifications after the container is located at terminal where payments are not supported. Orchestration keeps running normally for all supported features.
10023	Terminal not supported for payments	C	Returned by API response on orchestration update request if Guarantee Through Date is provided. Request is rejected.
10024	Terminal not supported for appointments	W	Returned by notifications after the container is located at terminal where appointments are not supported. Orchestration keeps running normally for all supported features.
10025	Terminal not supported for appointments	C	Returned by API response on orchestration update request if any appointment preferences are provided. Request is rejected.
10026	Unexpected error: reference code#[reference code]	C	Returned by API response and notifications whenever an unexpected error occurs. Manual support required.
10027	An orchestration for this container is active for another company	C	Returned by API response on orchestration creation request if the container is already orchestrated. Request is rejected.
10028	Terminal payments not supported. Only third party will be processed	W	Returned by notifications after the container is located at terminal where payments are not supported, except for third party such as PierPass & CleanTruck. Orchestration keeps running normally for all supported features.
10029	Orchestration update request accepted	I	Returned by API response on successful orchestration update request.
10030	Empty return not allowed by the terminal	W	Returned by notifications if the terminal does not support empty returns and empty return preferences were passed. Other orchestration features continue to execute normally.
10031	Updates cannot be passed during a reschedule request	W	Returned by API response on orchestration update request if more fields were passed other than time preferences. Only time preferences will be taken. Other fields will be ignored. Reschedule will be processed normally.
10032	Appointment not found in given location	C	Returned by API response on orchestration update or cancelation request if no appointment is found for that container.
10033	Appointment cannot be updated while awaiting response from internal appointment system	C	Returned by API response on orchestration update request if appointment data is passed and the system is waiting to receive acknowledgement of a previous appointment operation. Integrator should retry in this case.
10034	Container has moved to off-dock yard	I	Returned by notifications if the container has moved to an off-dock location.
10035	An empty container cannot be the primary in a dual move	C	Dual moves need to be implemented by adding an Empty on top of an existing pick appointment.
10036	This container has a dual move appointment.	C	The container requested as a dual is already in a dual orchestration.
10037	Not implemented yet	C	Not implemented yet.

10038	Invalid secondary container	C	Dual container does not exist or is not gated out yet. For Duals, only gated out empty containers are supported.
10039	No terminals identified to receive empty line type	C	Even though there are terminals supporting Empty reception, non of them accept this particular container.
m	Terminal not identified to receive empty line type	W	

Container Status Codes

Status Code	Status Description
N	Not Manifested
O	Manifested
AE	On Vessel
VA	Vessel Arrival
UV	Unload from Vessel
Y	In Yard
AL	On Rail
UR	Rail Unload
DN	Not Ready for Pickup
AV	Ready for Pickup
OA	Gate Out
I	Gate In

Container Activity Timeline Codes

Event Code	Event Description
CNMAN	Container Manifested
DISCH	Discharged
LFDAY	Last Free Day
PLAVI	Ready For Pickup
APNMT	Pregate
DPTRM	Departed Terminal
ARCUS	Arrived at Customer
EMTOT	Empty Released by Customer
EMTRC	Empty Received

Hold Reason Codes

Type	Code	Description
CUSTOMS	PQ	Has a Customs Hold
CUSTOMS	QQ	Customs Default Bill Hold
FREIGHT	FD	Freight is Due
OTHER	CF	CTF Hold
OTHER	TC	Held for Terminal Charges

OTHER	TM	TMF Hold
FEE	DEMARG	Demurrage

Note: codes not appearing on the above list could be returned. Integrators should build their logic to accept additional codes utilizing type and description.

Reservation Status

Status	Description
PENDING	Appointment pending confirmation or awaiting trucker SCAC assignment.
*CONFIRMED	Slot confirmed. Information required prior to execution. currently, only 'trucker_scac' and 'lic_plate_nbr' are required (see 'trucker_info' fields in 'Create orchestration object'). Lack of fee payment can also cause *CONFIRMED, so failure to extend Guarantee Through Date if it has been exceeded could also cause *CONFIRMED. Terminal Operating Systems could have other rules that could result in *CONFIRMED but these would not be due to lack of data passed. Clearance of Holds is one potential example.
CONFIRMED	Slot confirmed.
CANCELED	Appointment canceled.
COMPLETED	Appointment completed. Import Container left the facility or Empty Container arrived at the facility.
MISSED	Appointment missed (this could also mean it was completed outside the scheduled window – container gates out correctly).

Flows

Only Third-Party Payments

Terminal does not support any other functionality than third-party payment:

Action/Event	Orchestration Status	Container status	Notification
Orchestration submission received	active	NOT MANIFESTED	
Third-Party fees claimed	active	NOT MANIFESTED	
Container not found	stopped	N/A	
Third-Party notifies of collected fees	stopped	N/A	[<pre> { "fee_cd": "PIERPASS", "fee_desc": "TMF Fee", "fee_amt": "68.42", "fee_paid_dttm": "2023-01-01T00:00:00.0000000" }, { "fee_cd": "PIERPASS", "fee_desc": "CTF Fee", "fee_amt": "20.00", "fee_paid_dttm": "2023-01-01T00:00:00.0000000" }]</pre>