border express
committed to deliver 100%

Agent Integration Specification

Document History

Version	Date	Description	Author
0.1	1-Sep-2019	Document Creation	Nicole Brown
1.0	10-Sep-2018	Document Review	Wade Altmeier
1.1	16-Jul-2019	Updated Barcode Scans API Details	Nicole Brown
1.2	29-Jul-2020	Updated Authentication	Nicole Brown
1.3	27-Apr-2021	Updated Accepted Scan Types	Nicole Brown

Introduction

The purpose of this document is to provide technical information to Border Express' partners regarding the export and import of transportation data. Border Express currently exports data via XML and provides a REST API for data import (POD's, additional services, barcode and consignment events). The information contained in this document is solely for the intended recipient and is not to be shared or redistributed without the consent of Border Express.

Consignment Exports

There are two different export schedules running during the day, producing three different XML files. The XML structures should be the same, though not all nodes will be populated for all exports.

Export 1 (Manifested Consignment Export)

The first export is scheduled to occur every 30 minutes. It will export any con notes that have been added to a manifest for an export enabled agent, providing that the manifest is considered finalised for data entry.

A manifest exported in this way will only be exported once.

This export will generate the file "MANIFEST_{Manifest Number}.XML".

Export 2 (Non Manifested Consignment Export)

The second export is scheduled to occur every morning at 2:40am. It will export any con notes dated to either of the previous two working days, provided the agent is the preferred agent for the sender suburb (pickup) or receiver suburb (delivery) on the con note.

Any con notes exported in this way will only be exported once to the same agent for pickup, and once for delivery.

This export will generate the files "MANIFEST_{Manifest Number}_{Datestamp}.XML" for con notes that are on a manifest, and "NOMANIFEST_{Agent ID}_{Datestamp}.XML" for con notes that are not on a manifest.

FTP Transfer

Agents or their technical partners will provide Border Express with an FTP location, username and password and export files will be pushed to this location.

Imports

Border Express prefers to import data from all third parties using the Border Express REST API.

For setup and testing purposes please connect to our test API
https://integrationapi.borderexpress.com.au.

Full Border Express API documentation can be found at https://api.borderexpress.com.au. **This API should only be used after testing has been confirmed.**

The following API endpoints are available:

- POD (full image): Allows the creation of a POD from a provided image. A POD is usually created on job completion.
- POD (signature): Allows the creation of a Border Express POD from a provided signature. A POD is usually created on job completion.
- Barcode checkpoints: When an item barcode is scanned, a checkpoint event may need to be created in order to provide visibility on the item.
- Additional charges: Enables the creation of additional service notifications for a consignment. These are used to notify Border Express of any additional services performed for a customer at the point of pickup or delivery.
- Consignment events: Allows the submission of events against a consignment.
- Consignment status updates: Enables the details of a consignment on an agent manifest to be modified.

In order to request access to our API please email integrations@borderexpress.com.au and provide an email address to link your API account to.

API Authentication

Border Express uses OAuth 2.0 authorisation. Before you can start utilising the API you will need to contact Border Express by emailing **integrations@borderexpress.com.au**. Please provide an email address to link to your API account. After your account is set up, Border Express will send you a link to generate your client credentials, these will be used to retrieve an API bearer token.

To retrieve an API Token:

- Make a POST request to https://api.borderexpress.com.au/token with a header of: *"Content-Type: application/x-www-form-urlencoded"*.

The entered request body should contain your URL encoded credentials in the format of:

```
grant_type=client_credentials&client_id={clientId}&client_secret={clientSecret}
```

If successful a result will be returned in JSON:

```
{
    "access_token":"****",
    "token_type":"bearer",
    "expires_in":3599,
    "scope":"bex.webapi.basic",
    "userName":"username@borderexpress.com.au",
    ".issued":"Tue, 29 Aug 2017 05:45:16 GMT",
    ".expires":"Tue, 29 Aug 2017 06:45:16 GMT"
}
```

Using Your Token

To utilise your token, you must add the access token into the request header. When a token is first retrieved, it is valid for an hour.

In the request header, add the below values:

- Content-Type: application/json

- Authorization: bearer {access_token_value}

Tokens must be renewed once expired. If at token has been revoked by Border Express you will receive a 401 status code and will be required to re-authenticate.

POD (full image)

Allows the creation of a POD from a provided image. A POD is usually created on job completion. Only one of the image properties are required (Image or ImageBase64). Use Image if submitting multipart/formdata otherwise use ImageBase64.

Endpoint:

POST https://integrationapi.borderexpress.com.au/api/pods

Parameters:

pod (model, body):

The POD image that needs to be attached to the consignment.

Model:

```
PODImageCreateViewModel
{
        ManifestNumber (string, optional): The manifest number the POD is associated with,
        ConnoteNumber (string): The consignment number the POD is associated with,
        DateSigned (string): The date and time that the signature was signed,
        SignedBy (string): The name of the person who signed the proof,
        CollectedByUserId (integer, optional): The Id of the user that collected the proof,
        Image (HttpPostedFileMultipartModel, optional): The image contents,
        ImageBase64 (string, optional): The image content encoded as base64
}
HttpPostedFileMultipartModel
{
        FileContents (string, optional),
        FileName (string, optional),
        ContentType (string, optional)
}
```

Example:

```
{
  "ManifestNumber": "string",
  "ConnoteNumber": "string",
  "DateSigned": "2018-09-07T01:32:03.458Z",
  "SignedBy": "string",
  "CollectedByUserId": 0,
  "Image": {
          "FileContents": "string",
          "FileName": "string",
          "ContentType": "string"
  },
  "ImageBase64": "string"
}
```

POD (signature)

Allows the creation of a Border Express POD from a provided signature. A POD is usually created on job completion. Only one of the signature properties are required (Signature or SignatureBase64). Use Signature if submitting multipart/formdata otherwise use SignatureBase64.

Endpoint:

POST https://integrationapi.borderexpress.com.au/api/pods/signatures

Parameters:

signature (model, body):

The signature image that needs to be generated into a Border Express POD and attached to the consignment.

Model:

```
PODSignatureCreateViewModel
{
        ManifestNumber (string, optional): The manifest number the POD signature is associated with,
        ConnoteNumber (string): The consignment number the POD signature is associated with,
        DateSigned (string): The date and time that the signature was signed,
        SignedBy (string): The name of the person who signed the signature,
        CollectedByUserId (integer, optional): The Id of the user that collected the signature,
        Signature (HttpPostedFileMultipartModel, optional): The signature image,
        SignatureBase64 (string, optional): The signature image encoded as base64
}
HttpPostedFileMultipartModel
{
        FileContents (string, optional),
        FileName (string, optional),
        ContentType (string, optional)
}
```

Example:

```
{
  "ManifestNumber": "string",
  "ConnoteNumber": "string",
  "DateSigned": "2018-09-07T01:32:03.449Z",
  "SignedBy": "string",
  "CollectedByUserId": 0,
  "Signature": {
          "FileContents": "string",
          "FileName": "string",
          "ContentType": "string"
  },
  "SignatureBase64": "string"
}
```

Barcode scans

When an item barcode is scanned, a scan may need to be created in order to provide visibility on the item.

UserId and BranchId are only required when passing in a location type of 1.

Endpoint:

POST https://integrationapi.borderexpress.com.au/api/barcodes/{barcode}/scans

Parameters:

barcode (string, path):

The barcode number that was scanned.

scan (model, body):

The details of the scan.

Model:

```
BarcodeScanCreateViewModel
{
        ScanTypeId (integer),
        Barcode (string): The value of the barcode scanned ,
        UserId (integer, optional): The identity of the user that scanned the item ,
        ScanDate (string): The date and time the scan occurred ,
        LocationType (integer): Where the item was scanned ( -1 - Sender, -2 - Receiver, 1 - Branch ) ,
        BranchId (integer, optional): The identity of the branch where the item was scanned ,
        MethodId (integer, optional): The identity of the method used to perform the scan ( 1 - Physical (Scanned by a hardware
        scanner), 3 - Implied (Scan generated by software) ) ,
        ApplicationId (integer, optional): The identity of the application that performed the scan ,
        AreaId (integer, optional): The identity of the area the item was scanned in
}
```

Example:

```
{
 "ScanTypeId": 0,
 "Barcode": "string",
 "UserId": 0,
 "ScanDate": "2019-07-29T01:08:05.680Z",
 "LocationType": 0,
 "BranchId": 0,
 "MethodId": 0,
 "ApplicationId": 0,
 "AreaId": 0
}
```

Available Item Scans:

We currently allow the following item level scans to be submitted:

ID	Scan
-1	Checkpoint
3010	Load For Delivery
3011	Removed From Delivery
3020	Load For Onforward
3021	Removed From Onforward
2910	Hold Freight

Additional services

Enables the creation of additional service notifications for a consignment.

These are used to notify Border Express of any additional services performed for a customer at the point of pickup/delivery.

Endpoint:

POST https://integrationapi.borderexpress.com.au/api/connotes/{connoteNumber}/additionalServiceNotifications

Parameters:

connoteNumber (string, path):

The consignment the additional service is related to.

services (model, body):

A list of additional service notifications that need to be created on a consignment.

Model:

```
DeliveryAdditionalServiceNotificationsViewModel
{
        ManifestNumber (string, optional): The manifest the additional service is related too,
        ConnoteNumber (string): The consignment the additional service is related too, if provided in the URL the URL value will override the Body value,
        AdditionalServices (Array[AdditionalServiceNotificationViewModel]): A list of additional services performed
}
AdditionalServiceNotificationViewModel
{
        UserId (integer),
        ReasonId (integer),
        Qty (integer)
}
```

Example:

```
{
 "ManifestNumber": "string",
 "ConnoteNumber": "string",
 "AdditionalServices": [
        {
         "UserId": 0,
         "ReasonId": 0,
         "Qty": 0
        }
 ]
}
```

Available additional services:

We currently allow the following additional services to be submitted:

ID	Additional Service
134	Tailgate per Delivery.
139	Re-Delivery per Consignment.
144	Hand Unload per Consignment.

Consignment events

Allows the submission of events against a consignment.

Endpoint:

POST https://integrationapi.borderexpress.com.au/api/connotes/{connoteNumber}/Events

Parameters:

connoteNumber (string, path):

The number for the consignment to submit the event for.

events (model, body):

A list of events to be created for the consignment.

Model:

```
Array[CreateConnoteEventViewModel
{
            BranchId (integer): The Branch Id where the event occurred,
            UserId (integer): The User Id that performed the action,
            DateOccurred (string): The date and time that the event occurred, Supports ISO8601 formatted date/time,
            Code (integer): The unique event code,
            Comment (string, optional): Any additional information regarding the event
}]
```

Example:

```
[
 {
            "BranchId": 0,
            "UserId": 0,
            "DateOccurred": "2018-09-07T01:32:03.304Z",
            "Code": 0,
            "Comment": "string"
 }
]
```

Available events:

We currently allow the following events to be submitted:

Event Code	**Event**
70010000	In Depot
70011000	On Hold
70020000	Assigned
70021000	On Delivery
70022000	Delivered

Consignment status updates

Enables the details of a consignment on an agent manifest to be modified. Currently, only the "replace" PATCH specification operation is supported. Requires the consignments to be on a manifest.

Endpoint:

PATCH https://integrationapi.borderexpress.com.au/api/agentManifests/{manifestNumber}/consignments/{consignmentNumber}

Parameters:

manifestNumber (string, path):

The number of the agent runsheet the job is on.

consignmentNumber (string, path):

The consignment number of the job.

jsonPatch (model, body):

A list of instructions that specify what fields to change and the value to set.

Model:

```
PatchRequestModel
{
            Operations (Array[PatchDocumentModel], optional)
}
PatchDocumentModel
{
            OP (string, optional),
            Path (string, optional),
            Value (object, optional)
}
```

Example:

```
{
 "Operations": [
            {
             "OP": "string",
             "Path": "string",
             "Value": {}
            }
 ]
}
```

Supported Paths:

We currently allow the following paths to be updated:

Path	Value
/status	Completed, Rejected or OnboardForDelivery
/modifiedbyuserid	Integer
/modifiedbydriverid	Integer

/modifiedatbranchid	Integer
/modifiedstatusreasonid	Integer
/modifieddate	Datetime
/joborder	Integer
/location/longitude	Decimal
/location/latitude	Decimal

Available rejected statuses:

Please note that the rejected status is also used for deliveries that do not require a sign-on-glass.

We currently allow the following rejected statuses to be submitted:

ID	Completed Consignment Status
13	Partial delivery - MDA signed
9	Authority to leave
10	Paperwork signed

ID	Rejected Consignment Status
2	Excessive waiting time
3	Insufficient time
4	No forklift
6	Site closed – Card left
7	Vehicle too large to access site
8	Wrong address
11	Refused – Freight damaged
12	Refused – Freight short
13	Partial delivery - MDA signed
15	Tailgate required
16	Refused – Timeslot missed
21	Refused – Early or late delivery
22	Refused – Other

Production (Go Live)

Please note that all example/sample code is referencing the Border Express integration (testing) API. Once testing is completed and signed off on by a Border Express IT team member all code will need to be switched over to our production API https://api.borderexpress.com.au.

Manifested consignment export sample

This is the most common consignment manifest export sent to an agent when a consignment is manifest (and the manifest is marked as departed) in the Border Express systems.

Content of files "MANIFEST_{Manifest Number}.XML" and MANIFEST_{Manifest Number}_{Datestamp}.XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<Manifest>
        <ManifestNumber>2671533</ManifestNumber>
        <ManifestDate>31/08/2018</ManifestDate>
        <ManifestType>Onforward</ManifestType>
        <NumberOfConsignments>1</NumberOfConsignments>
        <ManifestFrom>Wagga Wagga</ManifestFrom>
        <ManifestTo>Wagga Wagga</ManifestTo>
        <DepartureDate>31/08/2018</DepartureDate>
        <DepartureTime>10:36:06 PM</DepartureTime>
        <ArrivalDate />
        <ArrivalTime />
        <Subcontractor>{Legal Name}</Subcontractor>
        <SubcontractorName>{Trading Name}</SubcontractorName>
        <SubcontractorAddress>123 Test Street</SubcontractorAddress>
        <SubcontractorSuburb>Wagga Wagga</SubcontractorSuburb>
        <SubcontractorState>NSW</SubcontractorState>
        <SubcontractorPostcode>2650</SubcontractorPostcode>
        <SubcontractorUserID>1234</SubcontractorUserID>
        <SubcontractorBranchID>3</SubcontractorBranchID>
        <Consignment>
                <ConsignmentNumber>5529184</ConsignmentNumber>
                <ConsignmentDate>30/08/2018</ConsignmentDate>
                <SenderName>Test Group</SenderName>
                <SenderStreetAddress>6 Bonville Ave</SenderStreetAddress>
                 <SenderStreetAddress1/>
                <SenderSuburb>Thornton</SenderSuburb>
                <SenderState>NSW</SenderState>
                <SenderPostcode>2322</SenderPostcode>
                <ReceiverName>Test Rec </ReceiverName>
                <ReceiverStreetAddress>Test Depot</ReceiverStreetAddress>
                <ReceiverStreetAddress1>3 William Street</ReceiverStreetAddress1>
                <ReceiverSuburb>Urana</ReceiverSuburb>
                <ReceiverState>NSW</ReceiverState>
                <ReceiverPostcode>2645</ReceiverPostcode>
                <ReceiverPhone />
                <ReceiverContact />
                <ReceiverReference />
                <Mode>Express</Mode>
                <FreightPayableBy>Test10</FreightPayableBy>
                <PalletType>N/A</PalletType>
                <SenderChepIn>0</SenderChepIn>
                <SenderChepOut>0</SenderChepOut>
                <ReceiverChepIn>0</ReceiverChepIn>
                <ReceiverChepOut>0</ReceiverChepOut>
                <SenderLoscamIn>0</SenderLoscamIn>
                <SenderLoscamOut>0</SenderLoscamOut>
                <ReceiverLoscamIn>0</ReceiverLoscamIn>
                <ReceiverLoscamOut>0</ReceiverLoscamOut>
                <PalletsTransferredDirect>No</PalletsTransferredDirect>
                <DeliveryChepTransferDocket />
                <DangerousGoods>No</DangerousGoods>
                 <SpecialInstructions>ATL</SpecialInstructions>
                 <PreferredSender>No</PreferredSender>
        <PreferredReceiver>Yes</PreferredReceiver>
        <LoadDetails>
            <ItemReference>N/A</ItemReference>
            <NumberOfUnits>1</NumberOfUnits>
            <UnitDescription>Pallet</UnitDescription>
            <Weight>600</Weight>
            <CubicUnits>1</CubicUnits>
            <Length>1.4</Length>
            <Width>1.2</Width>
            <Height>1.2</Height>
            <Cubic>2.016</Cubic>
            <CubicConverted>504</CubicConverted>
        </LoadDetails>
        <LoadDetails>
            <ItemReference>N/A</ItemReference>
            <NumberOfUnits>1</NumberOfUnits>
            <UnitDescription>Pallet</UnitDescription>
            <Weight>676</Weight>
            <CubicUnits>1</CubicUnits>
            <Length>1.4</Length>
            <Width>1.2</Width>
            <Height>1.2</Height>
            <Cubic>2.016</Cubic>
            <CubicConverted>504</CubicConverted>
        </LoadDetails>
        <CostOnforwardDetails>
            <ItemReference>N/A</ItemReference>
            <NumberOfUnits>1</NumberOfUnits>
            <UnitDescription>Pallet</UnitDescription>
            <Weight>600</Weight>
            <CubicUnits>1</CubicUnits>
            <Length>1.4</Length>
            <Width>1.2</Width>
            <Height>1.2</Height>
            <Cubic>2.016</Cubic>
            <CubicConverted>504</CubicConverted>
        </CostOnforwardDetails>
        <CostOnforwardDetails>
            <ItemReference>N/A</ItemReference>
            <NumberOfUnits>1</NumberOfUnits>
            <UnitDescription>Pallet</UnitDescription>
            <Weight>676</Weight>
            <CubicUnits>1</CubicUnits>
            <Length>1.4</Length>
            <Width>1.2</Width>
            <Height>1.2</Height>
            <Cubic>2.016</Cubic>
            <CubicConverted>504</CubicConverted>
        </CostOnforwardDetails>
        <LogisticUnitDetails>
            <Barcode>00093477800039666440</Barcode>
        </LogisticUnitDetails>
        <LogisticUnitDetails>
            <Barcode>00093477800039666457</Barcode>
        </LogisticUnitDetails>
        <DangerousGoodsDetails>
            <UNNumber />
            <Class />
            <ShippingName />
            <SubRisk />
            <PackagingGroup />
            <TypeOfPackage />
            <NumberOfPackages />
            <AggregateQuantity />
            <ContactName />
            <ContactPhoneNumber />
        </DangerousGoodsDetails>
    </Consignment>
</Manifest>
```

Non manifested consignment export sample

Content of file “NOMANIFEST_{Agent ID}_{Datestamp}.XML”.

```
<?xml version="1.0" encoding="UTF-8"?>
<Manifest>
        <ManifestType>Onforward</ManifestType>
        <NumberOfConsignments>1</NumberOfConsignments>
        <Subcontractor>{Legal Name}</Subcontractor>
        <SubcontractorName>{Trading Name}</SubcontractorName>
        <SubcontractorAddress>123 Test Street</SubcontractorAddress>
        <SubcontractorSuburb>Wagga Wagga</SubcontractorSuburb>
        <SubcontractorState>NSW</SubcontractorState>
        <SubcontractorPostcode>2650</SubcontractorPostcode>
        <SubcontractorUserID>1234</SubcontractorUserID>
        <SubcontractorBranchID>3</SubcontractorBranchID>
        <Consignment>
                <ConsignmentNumber>5529184</ConsignmentNumber>
                <ConsignmentDate>30/08/2018</ConsignmentDate>
                <SenderName>Test Group</SenderName>
                <SenderStreetAddress>6 Test Ave</SenderStreetAddress>
                <SenderSuburb>Thornton</SenderSuburb>
                <SenderState>NSW</SenderState>
                <SenderPostcode>2322</SenderPostcode>
                <ReceiverName>Test Rec </ReceiverName>
                <ReceiverStreetAddress>Test Depot</ReceiverStreetAddress>
                <ReceiverStreetAddress1>3 William Street</ReceiverStreetAddress1>
                <ReceiverSuburb>Urana</ReceiverSuburb>
                <ReceiverState>NSW</ReceiverState>
                <ReceiverPostcode>2645</ReceiverPostcode>
                <ReceiverPhone />
                <ReceiverContact />
                <ReceiverReference />
                <Mode>Express</Mode>
                <FreightPayableBy>TEST10</FreightPayableBy>
                <PalletType>N/A</PalletType>
                <SenderChepIn>0</SenderChepIn>
                <SenderChepOut>0</SenderChepOut>
                <ReceiverChepIn>0</ReceiverChepIn>
                <ReceiverChepOut>0</ReceiverChepOut>
                <SenderLoscamIn>0</SenderLoscamIn>
                <SenderLoscamOut>0</SenderLoscamOut>
                <ReceiverLoscamIn>0</ReceiverLoscamIn>
                <ReceiverLoscamOut>0</ReceiverLoscamOut>
                <PalletsTransferredDirect>No</PalletsTransferredDirect>
                <DeliveryChepTransferDocket />
                <DangerousGoods>No</DangerousGoods>
                <LoadDetails>
                        <ItemReference>N/A</ItemReference>
                        <NumberOfUnits>1</NumberOfUnits>
                        <UnitDescription>Pallet</UnitDescription>
                        <Weight>600</Weight>
                        <CubicUnits>1</CubicUnits>
                        <Length>1.4</Length>
                        <Width>1.2</Width>
                        <Height>1.2</Height>
                        <Cubic>2.016</Cubic>
                        <CubicConverted>504</CubicConverted>
                </LoadDetails>
                <LoadDetails>
                        <ItemReference>N/A</ItemReference>
                        <NumberOfUnits>1</NumberOfUnits>
                        <UnitDescription>Pallet</UnitDescription>
                        <Weight>676</Weight>
                        <CubicUnits>1</CubicUnits>
                        <Length>1.4</Length>
                        <Width>1.2</Width>
                        <Height>1.2</Height>
                        <Cubic>2.016</Cubic>
                        <CubicConverted>504</CubicConverted>
                </LoadDetails>
                <CostOnforwardDetails>
                        <ItemReference>N/A</ItemReference>
                        <NumberOfUnits>1</NumberOfUnits>
                        <UnitDescription>Pallet</UnitDescription>
                        <Weight>600</Weight>
                        <CubicUnits>1</CubicUnits>
                        <Length>1.4</Length>
                        <Width>1.2</Width>
                        <Height>1.2</Height>
                        <Cubic>2.016</Cubic>
                        <CubicConverted>504</CubicConverted>
                </CostOnforwardDetails>
                <CostOnforwardDetails>
                        <ItemReference>N/A</ItemReference>
                        <NumberOfUnits>1</NumberOfUnits>
                        <UnitDescription>Pallet</UnitDescription>
                        <Weight>676</Weight>
                        <CubicUnits>1</CubicUnits>
                        <Length>1.4</Length>
                        <Width>1.2</Width>
                        <Height>1.2</Height>
                        <Cubic>2.016</Cubic>
                        <CubicConverted>504</CubicConverted>
                </CostOnforwardDetails>
                <LogisticUnitDetails>
                        <Barcode>00093477800039666440</Barcode>
                </LogisticUnitDetails>
                <LogisticUnitDetails>
                        <Barcode>00093477800039666457</Barcode>
                </LogisticUnitDetails>
                <DangerousGoodsDetails>
                        <UNNumber />
                        <Class />
                        <ShippingName />
                        <SubRisk />
                        <PackagingGroup />
                        <TypeOfPackage />
                        <NumberOfPackages />
                        <AggregateQuantity />
                        <ContactName />
                        <ContactPhoneNumber />
                </DangerousGoodsDetails>
        </Consignment>
    </Manifest>
```

Sample API Code

POD (signature)

Verb: POST

URL: https://integrationapi.borderexpress.com.au/api/pods/signatures

Notes: CollectedByUserId is specific to Agent

Body:

```
{
"ManifestNumber": "{ManifestNumber}",
  "ConnoteNumber": "{ConnoteNumber}",
  "DateSigned": "2018-03-15T22:08:22.116Z",
  "SignedBy": "string",
  "CollectedByUserId": 1234,
  "SignatureBase64": "string"
}
```

Barcode scan

Verb: POST

URL: https://integrationapi.borderexpress.com.au/api/barcodes/{barcode}/scans

Notes: UserId, branchId is specific to Agent and are only required when passing in a location type of 1 (Branch)

Body:

```
{
  "ScanTypeId": -1, //Checkpoint
  "Barcode": "1234",
  "UserId": 1234,
  "ScanDate": "2018-09-13T12:00:00.000+10:00",
  "LocationType": 1,
  "BranchId": 10,
  "MethodId": 1, //Physical scan
  "ApplicationId": 0,
  "AreaId": 0
}
```

Consignment events

Verb: POST

URL: https://integrationapi.borderexpress.com.au/api/connotes/{ConnoteNumber}/events

Notes: BranchId and UserId are specific to Agent

Body:

```
[
{
"ConsignmentNumber": "{ConnoteNumber}",
"BranchID": "10",
"UserID": "1234",
"DateOccurred": "2018-09-13T12:00:00.000+10:00",
"Code": "70022000", //Delivered
"Comment": ""
}
]
```

Consignment status updates

On Board for Delivery

Verb: PATCH

URL:
https://integrationapi.borderexpress.com.au/api/agentManifests/{BexManifestNumber}/consignments/{ConnoteNumber}

Notes: modifiedbyuserid and modifiedatbranchid are specific to Agent

Body:

```
{
    "Operations": [{
        "OP": "replace",
        "Path": "/status",
        "Value": "OnBoardForDelivery"
    },
    {
        "OP": "replace",
        "Path": "/modifiedbyuserid",
        "Value": "1234"
    },
        {
        "OP": "replace",
        "Path": "/modifiedatbranchid",
        "Value": "10"
    },
    {
        "OP": "replace",
        "Path": "/modifieddate",
        "Value": "2018-03-14T13:47:05.4212620+10:00"
    },
    {
        "OP": "replace",
        "Path": "/location/longitude",
        "Value": "127.00"
    },
    {
        "OP": "replace",
        "Path": "/location/latitude",
        "Value": "81.23"
    }]
}
```

Delivery Complete

Verb: PATCH

URL:
https://integrationapi.borderexpress.com.au/api/agentManifests/{BexManifestNumber}/consignments/{ConnoteNumber}

Notes: modifiedbyuserid and modifiedatbranchid are specific to Agent

Body:

```
{
        "Operations": [{
                "OP": "replace",
                "Path": "/status",
                "Value": "completed"
        },
        {
                "OP": "replace",
                "Path": "/modifiedbyuserid",
                "Value": "1234"
        },
                {
                "OP": "replace",
                "Path": "/modifiedatbranchid",
                "Value": "10"
        },
        {
                "OP": "replace",
                "Path": "/modifieddate",
                "Value": "2018-03-14T13:47:05.4212620+10:00"
        },
        {
                "OP": "replace",
                "Path": "/location/longitude",
                "Value": "127.00"
        },
        {
                "OP": "replace",
                "Path": "/location/latitude",
                "Value": "81.23"
        }]
}
```

Delivery Rejected

Verb: PATCH

URL:

https://integrationapi.borderexpress.com.au/api/agentManifests/{BexManifestNumber}/consignments/{ConnoteNumber}

Notes: modifiedbyuserid and modifiedatbranchid are specific to Agent

Body:

```
{
        "Operations": [{
                "OP": "replace",
                "Path": "/status",
                "Value": "rejected"
        },
        {
                "OP": "replace",
                "Path": "/modifiedbyuserid",
                "Value": "1234"
        },
                {
                "OP": "replace",
                "Path": "/modifiedatbranchid",
                "Value": "10"
        },
        {
                "OP": "replace",
                "Path": "/modifiedstatusreasonid",
                "Value": “2” //Excessive waiting time
        },
        {
                "OP": "replace",
                "Path": "/modifieddate",
                "Value": "2018-03-14T13:47:05.4212620+10:00"
        },
        {
                "OP": "replace",
                "Path": "/location/longitude",
                "Value": "127.00"
        },
        {
                "OP": "replace",
                "Path": "/location/latitude",
                "Value": "81.23"
        }]
}
```