Avectous Connect API

Version 1.8.1.4

Important Information 2
Token API 5
Submit Feed API (Bulk) 8
Check Feed Submission Status 11
Get Shipment Confirmations 15
Get Receipt Confirmations 19
Get Inventory Adjustments 21
Get Inventory Snapshot (Legacy) 23
Check Order Status in WMS 25
Get Fulfillment Orders 29
Generate SSCC for a Fulfillment Order 31
Upload a Fufillment Shipment 35
Get Expected Receipts 41
Upload a Receipt Confirmation 43
Create an Order 47
Update Products in Bulk 52
Create a Bill Of Material 53
Create an Expected Receipt 58
Update an Expected Receipt 63
Cancel an Order 68
Cancel an Expected Receipt 73
Create a Work Order 78
Get Product Updates 83
Get Inventory Details (Current Snapshot View) 85

Important Information

API URL & Data Format

The base URL for Connect API is different for each customer and is provided by Avectous during integration onboarding. There are separate environments for sandbox (*connectdev*) and production (*connect*). The URL format is:

```
https://<SUBDOMAIN>.avectous.com/<APP_ENVIRONMENT>/api/
```

All API operations use JSON as the standard format for data interchange. Samples are available and is included with the documentation.

Authentication

All API users are assigned a username (*USER*) and password (*PASSWORD*) along with a set of claims which Avectous uses to authenticate the user. An access token is required in every API request and included in the Authorization HTTP header:

```
Authorization: Bearer <access_token>
```

To create an access token, call the Token API and provide the API credentials as part of the request header. This token needs to be regenerated after every 60 minutes.

Access

Avectous maintains a mapping of the user and company for which the user has authorized access. Most API methods require passing a Company code and/or Warehouse code in the request. This is validated against the mapping stored in Avectous systems to determine access to Connect API resources.

Integration Overview

Developers are provided sandbox API credentials to build and test your implementation. Some APIs require additional setup before it can be used. Work with your Avectous WMS implementation specialist to determine which integrations are needed.

Bulk vs. Transactional APIs

Avectous has a set of APIs which support either bulk or transactional data imports. The Feeds API is used for bulk data loading. Currently bulk feeds are only supported for Products, Sales Orders, and Expected Receipts.

The bulk Feed API is an asynchronous operation and provides a very efficient and effective way to load large data sets into Avectous. It is best suited for traditional scheduled, or batch data loads containing mass amounts of data and when the data is not expected to change often.

The transactional APIs process data in a synchronous manner and is meant for targeted, small loads of data. It's best suited for event-based data feeds where you'd like the information to be reflected immediately in Avectous. Most transactional APIs also provide better error details in the response compared to the bulk feeds and some operations are only available through transactional APIs e.g., cancelling an order.

Below is a typical workflow for sending data in bulk using feeds:

1. Token API

   Obtain an access token by calling the Token API with your username and password. The token will be used as the HTTP Authorization request header to authenticate a user agent with the server.

2. Submit Feed API

   Use this operation to submit data in bulk for asynchronous processing. When creating the JSON payload, ensure each message in the feed matches the feed type specified in the request (an invalid feed type returns an error). A FeedSubmissionId is returned in the API response and is used to identify a given transaction. This value can also be used with the Check Feed Status API to return the status of this transaction.

3. Check Feed Status

   You can use this operation to check the status of a feed submission transaction. The feedSubmissionId should match the value returned in the Submit Feed operation. It's highly recommended that users check the status using this API to ensure that transactions were processed successfully. If an error occurred, the error details are included in the response so you can correct the feed and resubmit it.

Third-Party WMS Integrations

A typical connection involves an integration with a customer's ERP system where Avectous WMS handles managing inventory, order fulfillments, and inbound receiving across multiple warehouses. In some cases, a

customer might have another warehouse using a different WMS but would like to have everything funnel through Avectous and back to their ERP system. This is referred to as a Third-Party WMS integration and is supported by Connect API.

Note: API operations intended only for Third-Party WMS integrations use are noted at the beginning of the API section.

Moving to Production

Once development is completed, it's recommended for customers to test their implementation internally before starting end-to-end testing in the sandbox environment. Avectous will assist with testing, but it is ultimately the responsibility of the customer to define the different scenarios and ensuring all business requirements are met.

Access to the production environment is usually granted a week or two prior to the targeted go-live date and only after testing is completed and all issues are resolved.

If you have any questions during the process, please contact either your WMS implementation specialist or Avectous support at support@avectous.com.

Token API

HTTP Request

GET https://<client_url>/connect/api/v1/token

Description

Authorization endpoint to obtain temporary access token. All clients are *required* to provide a valid, non-expired access token when making a request to all other Connect endpoints. Only pre-authorized users created by Avectous will be able to successfully generate a valid Token Response.

Request Headers

Name	Description
Accept	**Required.** application/json
USER	**Required**. Value provided by Avectous. Please refer to Important Information > Your Developer Account Info.
USERPASS	**Required**. Value provided by Avectous. Please refer to Important Information > Your Developer Account Info.

Token Response Element

Name	Data Type	Description
access_token	string	Token generated for authorized user. Use this in conjunction with value from *token_type* to set the request Authentication header.
token_type	string	The type of *access_token* used for authentication. **Value returned should be set to "Bearer".**
expires_in	long	Expiration time of the *access_token* in seconds since time of generation. **Token is set to expire after 1 hr.**

Error Response Element

Name	Data Type	Description
Message	string	Error occurred. Token Response was not generated.

Sample Request

```
curl -X GET https://.../connect/api/v1/token \
-H "Accept: application/json" \
-H "USER: YOUR_USERNAME" \
-H "USERPASS: YOUR_USERPASS" \
```

Sample Response

```
{
    “access_token”: “978789ecb22d11a4b4a8461582f9ba62”,
    “token_type”: ”Bearer”,
    “expires_in”: 3600
}
```

Submit Feed API (Bulk)

HTTP Request

POST https://<client_url>/connect/api/v1/feed?feedType={FeedTypeValue}

Description

This is to submit the feed of data and get a FeedSubmissionId response. The value of the FeedSubmissionId is in the form of a GUID.

Things to Consider

1. **Authorization:** Make sure to be using the token provided by the Token API.
2. Every message object in the feed must match the **FeedType** specified in the query parameter or the whole feed will fail.
3. For Product Update, please view the Product Update section for information.

Query Parameters

Name	Description
FeedType	**Valid values for FeedType:** a. SalesOrder b. ExpectedReceipt c. Product

Request Headers

Name	Description

Content-Type	application/json
Authorization	Bearer {token}

Request Elements

Name	Data Type	Description
ContentFeed	feedMessage[]	**Required**. The actual content of the feed itself, in JSON. You must include the **ContentFeed** in the body of the HTTP request. This will be an array of *feedMessage*(s). Please see *Sales Order* or *Expected Receipt* or *Product* files for specific definitions.

Sample Request

```
curl -X POST https://.../connect/api/v1/feed?feedType=SalesOrder \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-d '{FeedSubmissionJsonRequest}'
```

Sample Response

```
{
    “FeedSubmissionId”: “00000000-0000-0000-0000-000000000000”
}
```

Check Feed Submission Status

HTTP Request

GET https://<client_url>/connect/api/v1/feed?feedSubmissionId={id}

Description

To check the status of a submitted feed.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. The value for **FeedSubmissionId** is obtained from the Submit Feed API response.
3. If a feed has not finished processing, the response will only contain the Status and the ProcessResult of the feed will be null.

Query Parameters

Name	Description
FeedSubmissionId	Use the *FeedSubmissionId* response from the Submit Feed API.

Request Headers

Name	Description
Authorization	Bearer {token}

Response Elements

Name	Data Type	Description
FeedSubmissionId	Guid	**Required.**
Status	string	**Required. Possible Values:** a. Pending b. Processing c. Processed d. ProcessFailure
ProcessResult	ProcessResult	**Conditional.** Populated if *Status* returned is “Processed” or “ProcessFailure”, otherwise null.

ProcessResult Object

Name	Data Type	Description
TotalMessages	int	**Required.** Total number of messages received from the feed.
ProcessedMessages	int	**Required.** Number of messages that were successfully loaded from the original amount requested.

ErroredMessages	int	**Required**. Number of messages that were unable to load out of the original amount requested.
ErrorDetails	Error[]	**Conditional.** Populated for each line returned in *ErroredMessages*.

Error Object

Name	Data Type	Description
FeedMessageId	int	**Required.** Original *FeedMessageId* sent with the Feed Request.
ErrorMessage	string	**Required.**

Sample Request

```
curl -X GET https://.../connect/api/v1/feed?feedSubmissionId=00000000-0000-0000-0000-000000000000 \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
```

Sample Response

```
{
    "FeedSubmissionId": "63fed869-7db8-47fc-8c1c-c1c1cfc3c4b7",
```

```
    "Status": "Processed"
    "ProcessResult": [
        "TotalMessages": 4,
        "ProcessedMessages": 2,
        "ErroredMessages": 2,
        "ErrorDetails": [
            {
                "FeedMessageId": 1,
                "ErrorMessage": "Error Message 1"
            },
            {
                "FeedMessageId": 4,
                "ErrorMessage": "Error Message 4"
            }
        ]
    ]
}
```

Get Shipment Confirmations

HTTP Request

GET https://<client_url>/connect/api/v1/shipment?startDate={date}&endDate={date}&companyCode={string}&warehouseCode={string}

GET https://<client_url>/connect/api/v1/shipment?orderNumber={string}&companyCode={string}&warehouseCode={string}

Description

To retrieve information on shipment confirmations for a specific company, in a specific warehouse. Query for shipments during a time range or by order number. Filtering by order number returns all available shipments for a given order.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. **startDate** and **endDate** are both UTC Datetime with format yyyy-MM-ddThh:mm:ss
3. **orderNumber** should match the orderNumber value in the Sales Order.
4. You can choose to filter by **startDate**/**endDate** or **orderNumber**.

Query Parameters

Name	Data Type	Description
startDate	UTC Datetime	**Conditional.** Required if orderNumber not specified.
endDate	UTC Datetime	**Conditional.** Required if orderNumber not specified.

orderNumber	string	**Conditional.** Should match the given value in Sales Order.
companyCode	string	**Required.**
warehouseCode	string	**Required.**

Request Headers

Name	Description
Authorization	Bearer {token}

Response Elements

Name	Data Type	Description
Shipment	Shipment[]	A list of *Shipments*. Please see *Shipment* file for details.

Shipment Object

Name	Data Type	Description

ShipmentMasterContainers	ShipmentMasterContainer[]	Please see *Shipment* file > *ShipmentMasterContainer* tab for details.
ShipmentContainers	ShipmentContainer[]	Please see *Shipment* file > *ShipmentContainer* tab for details.
ShipmentOrders	ShipmentOrder[]	Please see *Shipment* file > *ShipmentOrder* tab for details.

ShipmentOrder Object

Name	Data Type	Description
ShipmentOrderItems	ShipmentOrderItem[]	A list of item details for a *Shipment Order*. Please see *Shipment* file > *ShipmentOrderItem* tab for details.
ShipmentOrderAddresses	ShipmentOrderAddresses[]	A list of addresses for a *Shipment Order*. Please see *Shipment* file > *ShipmentOrderAddress* tab for details.

Sample Request

```
curl -X GET https://.../connect/api/v1/shipment?startDate=2019-05-
20T19:04:24&endDate=2019-05-28T19:04:24&companyCode=Test&warehouseCode=1\
-H "Authorization: Bearer 123456" \
```

Get Receipt Confirmations

HTTP Request

GET https://<client_url>/connect/api/v1/receipt?startDate={date}&endDate={date}&companyCode={string}&warehouseCode={string}

Description

To retrieve information on receipts by specific date ranges, company, and warehouse.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. **startDate** and **endDate** are both UTC Datetime with format yyyy-MM-ddThh:mm:ss

Query Parameter

Name	Data Type	Description
startDate	UTC Date	**Required.**
endDate	UTC Date	**Required.**
companyCode	string	**Required.**
warehouseCode	string	**Required.**

Request Headers

Name	Description
Authorization	Bearer {token}

Response Elements

Name	Data Type	Description
Receipts	Receipts[]	A list of *Receipts*. Please see *ReceiptConfirmation* JSON file for definitions.

Receipt Object

Name	Data Type	Description
ReceiptDetail	ReceiptDetail[]	A list of *ReceiptDetails* per *Receipt*. Please see *ReceiptConfirmation* file > *ReceiptItems* Tab for definitions.

Sample Request

```
curl -X GET https://.../connect/api/v1/receipt?startDate=2019-05-
20T19:04:24&endDate=2019-05-28T19:04:24&companyCode=Test&warehouseCode=1\
-H "Authorization: Bearer 123456" \
```

Get Inventory Adjustments

HTTP Request

GET https://<client_url>/connect/api/v1/inventoryAdjustment?startDate={date}&endDate={date}&companyCode={string}&warehouseCode={string}

Description

To retrieve information on inventory adjustments by specific date ranges, company, and warehouse.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. **startDate** and **endDate** are both UTC Datetime with format yyyy-MM-ddThh:mm:ss

Query Parameter

Name	Data Type	Description
startDate	UTC Date	**Required.**
endDate	UTC Date	**Required.**
companyCode	string	**Required.**
warehouseCode	string	**Required.**

Request Headers

Name	Description
Authorization	Bearer {token}

Response Elements

Name	Data Type	Description
InventoryAdjustments	InventoryAdjustment[]	A list of *Inventory Adjustments*. Please see *InventoryAdjustment* JSON file for definitions.

Sample Request

```
curl -X GET https://.../connect/api/v1/inventoryAdjustments?startDate=2019-05-
20T19:04:24&endDate=2019-05-28T19:04:24&companyCode=Test&warehouseCode=1\
-H "Authorization: Bearer 123456" \
```

Get Inventory Snapshot (Legacy)

Note: this method is deprecated and will no longer receive support updates.

HTTP Request

GET https://<client_url>/connect/api/v1/inventorysnapshot?companyCode={string}&warehouseCode={string}&zone={string}

Description

To retrieve an inventory snapshot report by company, warehouse, and zone (optional). Note: available only for clients using older version of Connect API. For new integrations, use the Inventory Detail API instead.

Things to Consider

1 **Authorization:** Provide the access token returned by the Token API.

Query Parameter

Name	Data Type	Description
companyCode	string	**Required.**
warehouseCode	string	**Required.**
zone	string	**Optional.**

Request Headers

Name	Description
Authorization	Bearer {token}

Response Elements

Name	Data Type	Description
InventorySnapshot	InventorySnapshot[]	A list of *InventorySnapshots* items. Please see *Snapshot* file for definitions.

Sample Request

```
curl -X GET
https://.../connect/api/v1/inventorysnapshot?companyCode=Test&warehouseCode=1&zone=RECE
IVING\
-H "Authorization: Bearer 123456" \
```

Check Order Status in WMS

HTTP Request

POST https://<client_url>/connect/api/v1/orderstatus?company={string}&warehouse={string}

Description

To check the status of the order(s) in WMS.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. For orders created via bulk feeds, it's recommended to wait at least 30 minutes before calling this operation.

Request Headers

Name	Description
Content-Type	application/json
Authorization	Bearer {token}
Accept	application/json

Request Elements

Name	Data Type	Description

Content Body	string	**Required**. Comma-separated string of *OrderNumbers* e.g., "14513,14267,13023"

Response Elements

Name	Data Type	Description
Object	OrderStatus[]	An array of *OrderStatus*.

OrderStatus Object

Name	Data Type	Description
OrderNumber	string	Requested *OrderNumber*.
Status	string	*Status* for the order. Possible status types returned: • ***Un-Waived*** – Pending action. • ***Waived*** – Waiting for picker to be assigned. • ***Picking*** – Picker assigned. • ***Packed*** – Manifested with printed label ("Ready to Ship"). • ***Shipped*** – Container has closed (order leaving/left facility).

		• ***Error*** – no existing record found for this order number.
DateChanged	UTC Date	**Nullable**. Timestamp of when order was last updated.

Sample Request

```
 curl -X POST
https://.../connect/api/v1/orderstatus?company={companyName}&warehouse={warehouseName}
\
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-H "Accept: application/json" \

-d “14513,14267,13023”
```

Sample Response

```
[
    {
        "OrderNumber": "14513",
        "Status": "Shipped",
        "DateChanged": "2019-09-05T01:59:00.42"
    },
    {
        "OrderNumber": "14267",
        "Status": "Shipped",
        "DateChanged": "2019-09-04T17:26:41.85"
    },
    {
        "OrderNumber": "13023",
```

```
        "Status": "Waived",
        "DateChanged": "2019-09-11T15:10:12.33"
    }
]
```

Get Fulfillment Orders

Note: this method is intended for Third-Party WMS integration types.

HTTP Request

GET https://<client_url>/connect/api/v1/fulfillmentorder?startDate={date}&endDate={date}

Description

To retrieve information on fulfillment orders by specific date ranges. This method is intended for use by Third-Party WMS integrators. Use this method to pull new orders ready for fulfillment. Once an order is fulfilled by the Third-Party WMS system, upload the shipment to Avectous by calling Upload a Fulfillment Shipment.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. **startDate** and **endDate** are both UTC Datetime with format yyyy-MM-ddThh:mm:ss
3. This method is for Third-Party WMS integration types.

Query Parameter

Name	Data Type	Description
startDate	UTC Date	**Required.**
endDate	UTC Date	**Required.**

Request Headers

Name	Description
Authorization	Bearer {token}

Response Elements

Name	Data Type	Description
FulfillmentOrder	FulfillmentOrder[]	A list of *FulfillmentOrders*. Please see *FulfillmentOrder* file for definitions.

Sample Request

```
curl -X GET https://.../connect/api/v1/fulfilmentorder?startDate=2019-12-
12T05:30:03&endDate=2019-12-12T05:50:49\
-H "Authorization: Bearer 123456" \
```

Generate SSCC for a Fulfillment Order

Note: this method is intended for Third-Party WMS integration types.

HTTP Request

GET https://<client_url>/connect/api/v1/fulfillmentordersscc?
fulfillmentChannelOrderHeaderId={int}&forceRegeneration={boolean}

Description

To retrieve a list of SSCC codes generated for the specified fulfillment order. This method requires additional setup before it can be used. If an integration is needed, please contact Avectous for more details.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. The **fulfillmentChannelOrderHeaderId** should match an Id returned from the Get Fulfillment Orders endpoint.
3. **forceRegeneration** is always set to *true* on the first request for a given fulfillment order.

Query Parameter

Name	Data Type	Description
fulfillmentChannelOrderHeaderId	int	**Required.** Unique Identifier of a *FufillmentOrder*.
forceRegeneration	Boolean	**Required.** Possible values: • ***true*** – a new set will be generated and returned.

		• ***false*** – a previously generated SSCCs will be returned.

Request Headers

Name	Description
Authorization	Bearer {token}

Response Elements

Name	Data Type	Description
FulfillmentChannelOrderHeaderId	int	Unique identifier of the requested *FulfillmentOrder*.
SSCCs	SSCC[]	A list of *SSCCs*. Please view *SSCC* Object for additional mapping info.

SSCC Object

Name	Data Type	Description
SSCC	string	Serial Shipping Container Code for the container or pallet.

Contents	Content[]	A list of *Contents* for that *SSCC*. Please view *Content* Object for additional mapping info.

Content Object

Name	Data Type	Description
FulfillmentChannelItemNumber	string	Product identifier for the items in this unit.
Quantity	int	Number of items in this unit.
Reference1	string	Any additional info will be populated here.
Reference2	string	Any additional info will be populated here.
Reference3	string	Any additional info will be populated here.

Sample Request

```
curl -X GET https://.../connect/api/v1/
FulfillmentOrderSSCC?fulfillmentChannelOrderHeaderId=12&forceRegeneration=true \
-H "Authorization: Bearer 123456" \
```

Sample Response


```
{
  "FulfillmentChannelOrderHeaderId": 57529,
  "SSCCs": [
    {
      "SSCC": "00084004600085027662",
      "Contents": [
        {
          "FulfillmentChannelItemNumber": "32S321",
          "Quantity": 1,
          "Reference1": null,
          "Reference2": null,
          "Reference3": null
        }
      ]
    }
  ]
}
```

Upload a Fufillment Shipment

Note: this method is intended for Third-Party WMS integration types.

HTTP Request

POST https://<client_url>/connect/api/v1/shipment

Description

To submit a shipment for existing fulfillment order(s) and get a shipment status response. The value of the ShipmentId is in the form of a GUID.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. For a list of status codes and descriptions, please see Error Codes.

Request Headers

Name	Description
Content-Type	application/json
Authorization	Bearer {token}
Accept	application/json

Request Elements

Name	Data Type	Description

Content Body	ShipmentObject	**Required**. Please see *Shipment* file for definitions.

Response Elements

Name	Data Type	Description
Status	string	Processing result for the shipment. Possible status codes returned: • ***Success*** – Shipment created. • ***Failure*** – failed to create Shipment. Please see Error Codes for a list of potential errors.
StatusDetails	string	Description of the status.
ShipmentId	Guid	Unique identifier of the processed shipment. This value can be used to reference the shipment later. *Note*: *a failed shipment will return "00000000-0000-0000-0000-000000000000"*

Sample Request

```
curl -X POST https://.../connect/api/v1/shipment \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-H "Accept: application/json" \

-d {Shipment Object}
```

Sample Response

```
{
    “Status”: “Failure”,
    “StatusDetails”: “Invalid Quantity”,
    “ShipmentId”: “00000000-0000-0000-0000-000000000000”
}
```

Error Codes

Error	Description
Missing ShipmentContainers	Shipment does not contain any ShipmentContainers.
Missing ShipmentOrders	Shipment does not contain any ShipmentOrders.
Missing ShipmentOrderAddresses	ShipmentOrder must contain at least one ShipmentOrderAddress.
Missing ShipmentOrderItems	ShipmentOrder must contain at least one ShipmentOrderItem.

Invalid NumberOfCartons	NumberOfCartons in Shipment is missing or does not match number of ShipmentContainer elements.
Invalid NumberOfPallets	NumberOfPallets in Shipment does not match number of ShipmentMasterContainer elements.
Missing MasterContainer LP	One or more LP(s) are missing in ShipmentMasterContainers.
Missing MasterContainer SSCC	One or more SSCC(s) are missing in ShipmentMasterContainers.
Missing MasterContainer TrackingNumber	One or more TrackingNumber(s) are missing in ShipmentMasterContainers.
Duplicate MasterContainer LP	ShipmentMasterContainers contains duplicate LPs.
Invalid MasterContainer ParentLP	One or more ParentLP(s) are invalid in ShipmentMasterContainers.
Invalid MasterContainer TotalUnits	TotalUnits in ShipmentMasterContainer is missing or does not match number of associated ShipmentOrderItems
Missing Container LP	One or more LP(s) are missing in ShipmentContainers.
Missing Container SSCC	One or more SSCC(s) are missing in ShipmentContainers.
Missing Container TrackingNumber	One or more TrackingNumber(s) are missing in ShipmentContainers.

Duplicate Container LP	ShipmentContainers contains duplicate LPs.
Invalid Container ParentLP	One or more ParentLP(s) are invalid in ShipmentContainers.
Invalid Container TotalUnits	TotalUnits in ShipmentContainer is missing or does not match number of associated ShipmentOrderItems
Invalid Warehouse	One or more Warehouse(s) are invalid in ShipmentOrders.
Invalid Company	One or more Company(s) are invalid in ShipmentOrders.
Invalid OrderNumber	One or more OrderNumber(s) are invalid in ShipmentOrders.
Missing Address Element	One or more of the required fields in ShipmentOrderAddresses are missing.
Invalid AddressType	One or more AddressType(s) are invalid in ShipmentOrderAddresses.
Invalid Line	One or more Line(s) are invalid in ShipmentOrderItems.
Invalid LP	One or more LP(s) are invalid in ShipmentOrderItems.
Invalid ProductId	One or more ProductId(s) are invalid in ShipmentOrderItems.
Invalid Quantity	Quantity must be greater than 0 in ShipmentOrderItems.

Order Line Quantity Limit Exceeded	One or more Shipped Line Quantities exceeds original Order Item Quantity.
Duplicate Shipment	System found duplicate Shipment request for FulfillmentChannelShipmentId {0}.
Invalid Number Value	One or more numeric field(s) are invalid.
ProcessFailure	Failed adding record to Shipment table for Shipment.

Get Expected Receipts

Note: this method is intended for Third-Party WMS integration types.

HTTP Request

GET https://<client_url>/connect/api/v1/expectedreceipt?startDate={date}&endDate={date}&receiptType={string}

Description

To retrieve information on expected receipts by specific date ranges and receipt type.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. **startDate** and **endDate** are both UTC Datetime with format yyyy-MM-ddThh:mm:ss

Query Parameter

Name	Data Type	Description
startDate	UTC Date	**Required.**
endDate	UTC Date	**Required.**
receiptType	string	**Required.** Possible valid values for *receiptType*: *PO, RA, TO*

Request Headers

Name	Description
Authorization	Bearer {token}

Response Elements

Name	Data Type	Description
ExpectedReceipt	ExpectedReceipt[]	A list of *ExpectedReceipts*. Please see *ExpectedReceipt* file for definitions.

Sample Request

```
curl -X GET https://.../connect/api/v1/expectedreceipt?startDate=2019-12-12T05:30:03&endDate=2019-12-12T05:50:49\
-H "Authorization: Bearer 123456" \
```

Upload a Receipt Confirmation

Note: this method is intended for Third-Party WMS integration types.

HTTP Request

POST https://<client_url>/connect/api/v1/receipt

Description

To submit a receipt confirmation for an expected receipt and get a receipt status response. The value of the *ReceiptId* is in the form of a GUID.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. For a list of status codes and descriptions, please see Error Codes.

Request Headers

Name	Description
Content-Type	application/json
Authorization	Bearer {token}
Accept	application/json

Request Elements

Name	Data Type	Description
Content Body	ReceiptObject	**Required**. Please see *Receipt* file for definitions.

Response Elements

Name	Data Type	Description
Status	string	Processing result for the receipt. Possible status codes returned: • ***Success*** – Receipt created. • ***Failure*** – failed to create Receipt. Please see Error Codes for a list of potential errors.
StatusDetails	string	Description of the status.
ReceiptId	Guid	Unique identifier of the processed receipt. This value can be used to reference the receipt later. *Note*: *a failed receipt will return "00000000-0000-0000-0000-000000000000"*

Sample Request

```
curl -X POST https://.../connect/api/v1/receipt \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-H "Accept: application/json" \

-d {Receipt Object}
```

Sample Response

```
{
    “Status”: “Failure”,
    “StatusDetails”: “Missing Lines”,
    “ShipmentId”: “00000000-0000-0000-0000-000000000000”
}
```

Error Codes

Error	Description
Missing ReceiptItems	Receipt does not contain any ReceiptItems.
Invalid ProductId	One or more ProductId(s) are invalid in ReceiptItems.
Invalid Quantity	Quantity must be greater than 0 in ReceiptItems.
Invalid Line Numbers	One or more Line(s) are invalid in ReceiptItems.
Invalid Warehouse	Warehouse is invalid in Receipt.

Invalid Company	Company is invalid in Receipt.
Invalid Line Numbers	One or more Line(s) are invalid in ReceiptItems.
Expected Receipt Id does not exist	ExpectedReceiptId is invalid in Receipt.
Duplicate Receipt	System found duplicate Receipt request for FulfillmentChannel ReceiptId {0}.

Create an Order

HTTP Request

POST https://<client_url>/connect/api/v1/order

Description

To submit an order and get a status response for the processed order in a synchronous manner.

Things to Consider

1 **Authorization:** Provide the access token returned by the Token API.

Request Headers

Name	Description
Content-Type	application/json
Authorization	Bearer {token}
Accept	application/json

Request Elements

Name	Data Type	Description

Content Body	OrderObject	**Required**. Please see *Order* JSON file for definitions.

Response Elements

Name	Data Type	Description
LogId	Guid	Unique identifier for the request.
Status	string	Processing result for the *Order*. Possible status codes returned: • ***Success*** – order created. • ***Error*** – order could not be created. • ***Warning*** – order created but some data could not be verified.
ReceiptId	long	Unique identifier for the processed order. *Note*: *a failed order will return as null.*
Messages	Message[]	A list of status messages for errors/warnings. *Note: a successful order will return an empty list.*

Message Object

Name	Data Type	Description
Code	string	Status code for the raised error or warning.
Context	string	Order processing step where error or warning was raised. Possible values returned: • ***Validate*** – error occurred during staging validation. • ***Transform*** – error occurred while processing validated order. • ***Fulfill*** – error occurred during order fulfillment. • ***Channel Upload*** – order was rejected by fulfillment channel.
Severity	string	Level of impact for the raised error or warning. Possible severity levels returned in decreasing order: • ***Fatal*** – critical error occurred leading to unexpected/severe results. • ***Warning*** – event occurred which may result in potential errors

Message	string	Description of error or warning event that occurred.

Sample Request

```
curl -X POST https://.../connect/api/v1/order \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-H "Accept: application/json" \

-d {Order Object}
```

Sample Response

```
{
    "LogId": " 443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Success",
    "ReceiptId": 4793137,
    "Messages": []
}
```

Sample Error Response

```
{
    "LogId": " bd3e35c2-d246-46cf-9af8-25ba0fe84ca8",
    "Status": "Failure",
    "ReceiptId": null,
    "Messages": [
        {
```

```
            "Code": "SO005",
            "Context": "Validate",
            "Severity": "Fatal",
            "Message": "Duplicate SalesChannelOrderNumber '87095' in SalesOrder_Header"
        }
    ]
}
```

Update Products in Bulk

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. **Set Up:** Make sure the client already configured which fields to update via Connect UI prior to sending a product. Please refer to “How To Configure Product Update Controls” additional document.

How To Send Product Update

- Product update can be treated as if you are sending a regular product to the API.
- Please refer to Submit Feed API to send a Product.
- The only difference is you can send the same *ProductId* but include different values for other columns.

Create a Bill Of Material

HTTP Request

POST https://<client_url>/connect/api/v1/billOfMaterial

Description

To submit a bill of material (BOM).

Things to Consider

1 **Authorization:** Provide the access token returned by the Token API.

Request Headers

Name	Description
Content-Type	application/json
Authorization	Bearer {token}
Accept	application/json

Request Elements

Name	Data Type	Description

Content Body	BillOfMaterialObject	**Required**. Please see *BillOfMaterial* JSON file for definitions.

Response Elements

Name	Data Type	Description
LogId	Guid	Unique identifier for the request.
Status	string	Processing result for the BOM request. Possible status codes returned: • ***Success*** – bill of material is downloaded. • ***Error*** – bill of material could not be downloaded.
ReceiptId	long	Unique identifier for the processed BOM. *Note*: *a failed BOM will return as null*
Messages	Message[]	A list of status messages for errors/warnings. *Note: a successful bill of material will return an empty list.*

Message Object

Name	Data Type	Description
Code	string	Status code for the raised error or warning.
Context	string	Processing step where error or warning was raised. Possible values returned: • ***Validate*** – error occurred while validating the bill of material request • ***Channel Upload*** – error occurred while sending bill of material request to the fulfillment channel system
Severity	string	Level of impact for the raised error or warning. Possible severity levels returned in decreasing order: • ***Fatal*** – critical error occurred leading to unexpected/severe results • ***Warning*** – event occurred which may result in potential errors

Message	string	Description of error or warning event that occurred.

Sample Request

```
curl -X POST https://.../connect/api/v1/billOfMaterial \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-H "Accept: application/json" \

-d {Bill of Material Object}
```

Sample Successful Response

```
{
    "LogId": " 443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Success",
    “ReceiptId” : 63,
    "Messages": []
}
```

Sample Error Response

```
{
    "LogId": "443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Error",
    “ReceiptId” : null,
    "Messages": [
        {
            "Code": "BOM006",
```

```
            "Context": " Validate ",
            "Severity": "Fatal",
            "Message": "Missing Staging_Product_BillOfMaterial_Components”
        }
    ]
}
```

Create an Expected Receipt

HTTP Request

POST https://<client_url>/connect/api/v1/expectedreceipt

Description

To submit an expected receipt and get a status response for the processed expected receipt in a synchronous manner.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. List of possible valid expected receipt types: PO, RA, TO

Request Headers

Name	Description
Content-Type	application/json
Authorization	Bearer {token}
Accept	application/json

Request Elements

Name	Data Type	Description

Content Body	ExpectedReceiptObject	**Required**. Please see *ExpectedReceipt* JSON file for definitions.

Response Elements

Name	Data Type	Description
LogId	Guid	Unique identifier for the request.
Status	string	Processing result for the expected receipt. Possible status codes returned: • ***Success*** – expected receipt created. • ***Error*** – expected receipt could not be created. • ***Warning*** – expected receipt created but some data could not be verified.
ReceiptId	long	Unique identifier for the processed expected receipt. *Note*: *a failed expected receipt will return as null*
Messages	Message[]	A list of status messages for errors/warnings.

		Note: a successful expected receipt will return an empty list.

Message Object

Name	Data Type	Description
Code	string	Status code for the raised error or warning.
Context	string	Expected receipt processing step where error or warning was raised. Possible values returned: • ***Validate*** – error occurred during staging validation. • ***Transform*** – error occurred while processing validated expected receipt. • ***Channel Upload*** – expected receipt was rejected by fulfillment channel.
Severity	string	Level of impact for the raised error or warning. Possible severity levels returned in decreasing order: • ***Fatal*** – critical error occurred leading to unexpected/severe results.

		• ***Warning*** – event occurred which may result in potential errors.
Message	string	Description of error or warning event that occurred.

Sample Request

```
curl -X POST https://.../connect/api/v1/expectedreceipt \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-H "Accept: application/json" \

-d {ExpectedReceipt Object}
```

Sample Response

```
{
    "LogId": " 443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Success",
    "ReceiptId": 4793137,
    "Messages": []
}
```

Sample Error Response

```
{
    "LogId": " bd3e35c2-d246-46cf-9af8-25ba0fe84ca8",
    "Status": "Error",
```

```
    "ReceiptId": null,
    "Messages": [
        {
            "Code": "ER009",
            "Context": "Validate",
            "Severity": "Fatal",
            "Message": "ExpectedQuantity must be greater than 0"
        }
    ]
}
```

Update an Expected Receipt

HTTP Request

PUT https://<client_url>/connect/api/v1/expectedreceipt

Description

To update an existing expected receipt and get a status response for the processed result in a synchronous manner. An existing expected receipt is based on the values provided in the Company, Warehouse, ER Type, ER Number fields (and Container Number if permitted) during initial ER creation. For details on how to create an expected receipt, see Submit Feed or Create Expected Receipt. With the addition of ER Update Controls, the ER can be configured to allow updates on some of the available header fields.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. List of possible valid expected receipt types: PO, RA, TO
3. By default, all lines for an existing ER get deleted by the system and are replaced by the new lines provided in the update request.
4. The ER Update Controls is made available through InnovConnect UI. For more details, please consult with your implementation specialist to have it set up for your use.
5. *(**Applies only to PO**)* To update an existing expected receipt with multiple containers against different lines, the ER needs to first be created without a container number specified. Otherwise, you must use Post Expected Receipt to create the ER with the specified container number and associated lines. *Note: ER Update Controls must have updates disabled on ContainerNumber for this to work.*

Request Headers

Name	Description

Content-Type	application/json
Authorization	Bearer {token}
Accept	application/json

Request Elements

Name	Data Type	Description
Content Body	ExpectedReceiptObject	**Required**. Please see *ExpectedReceipt* JSON file for definitions.

Response Elements

Name	Data Type	Description
LogId	Guid	Unique identifier for the request.
Status	string	Processing result for the *ExpectedReceipt*. Possible status codes returned: • ***Success*** – expected receipt created.

		• ***Error*** – failed to create expected receipt. • ***Warning*** – expected receipt created but some data could not be verified.
Messages	Message[]	A list of status messages for errors/warnings. *Note: a successful expected receipt will return an empty list.*

Message Object

Name	Data Type	Description
Code	string	Status code for the raised error or warning.
Context	string	Expected receipt processing step where error or warning was raised. Possible values returned: • ***Validate*** – error occurred during staging validation. • ***Channel Upload*** – expected receipt was rejected by fulfillment channel. • ***Update*** – error occurred while processing validated expected receipt.

Severity	string	Level of impact for the raised error or warning. Possible severity levels returned in decreasing order: • ***Fatal*** – critical error occurred leading to unexpected/severe results. • ***Warning*** – event occurred which may result in potential errors.
Message	string	Description of error or warning event that occurred.

Sample Request

```
curl -X PUT https://.../connect/api/v1/expectedreceipt \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-H "Accept: application/json" \

-d {ExpectedReceipt Object}
```

Sample Response

```
{
    "LogId": " 443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Success",
    "Messages": []
```

```
}
```

Sample Error Response

```
{
    "LogId": " bd3e35c2-d246-46cf-9af8-25ba0fe84ca8",
    "Status": "Error",
    "Messages": [
        {
            "Code": "PSE",
            "Context": "Pre-Store Validation",
            "Severity": "Fatal",
            "Message": "No Expected Receipt was found."
        }
    ]
}
```

Cancel an Order

HTTP Request

PUT https://<client_url>/connect/api/v1/order/{orderId}/cancel

Description

To cancel an existing sales order created through the Order API. Depending on the status of the order in the WMS, an order may or may not be eligible for cancellation. For example, an order may not be cancelled if it has already been picked, packed, or shipped.

Things to Consider

1 **Authorization:** Provide the access token returned by the Token API.

Query Parameters

Name	Description
orderId	Use the *ReceiptId* returned from the Order API response.

Request Headers

Name	Description
Content-Type	application/json

Authorization	Bearer {token}
Accept	application/json

Request Elements

Name	Data Type	Description
Content Body	CancelOrderRequest	**Required**. See *Cancel Order Request* below for more details.

Cancel Order Request

Name	Data Type	Description
Company	string	**Required**. Seller company code. Unique code issued by Avectous.
OrderId	int	**Required.** Identifier for the sales order to cancel. See *ReceiptId* from Order API response.
ReasonCode	string	**Optional.** Reason for cancellation.

Response Elements

Name	Data Type	Description
LogId	Guid	Unique identifier for the request.
Status	string	Processing result for the order cancellation request. Possible status codes returned: • ***Success*** – order cancelled. • ***Error*** – order could not be cancelled.
Messages	Message[]	A list of status messages for errors/warnings. *Note: a successful cancellation will return an empty list.*

Message Object

Name	Data Type	Description
Code	string	Status code for the raised error or warning.
Context	string	Processing step where error or warning was raised.

		Possible values returned: • ***Validate*** – error occurred while validating the cancel order request. • ***Channel Upload*** – error occurred while sending cancel order request to the fulfillment channel system.
Severity	string	Level of impact for the raised error or warning. Possible severity levels returned in decreasing order: • ***Fatal*** – critical error occurred leading to unexpected/severe results. • ***Warning*** – event occurred which may result in potential errors.
Message	string	Description of error or warning event that occurred.

Sample Request

```
curl -X PUT https://.../connect/api/v1/order/12345/cancel \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-H "Accept: application/json" \

-d {CancelOrderRequest Object}
```

Sample Response


```
{
    "LogId": " 443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Success",
    "Messages": []
}
```


Sample Error Response


```
{
    "LogId": "443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Error",
    "Messages": [
        {
            "Code": "IO025",
            "Context": "Validate",
            "Severity": "Fatal",
            "Message": "Order cannot be cancelled due to warehouse status"
        }
    ]
}
```

Cancel an Expected Receipt

HTTP Request

PUT https://<client_url>/connect/api/v1/expectedreceipt/{expectedReceiptId}/cancel

Description

To cancel an existing expected receipt created through the Expected Receipt API. Depending on the status of the ER in the WMS, an ER may or may not be eligible for cancellation.

Things to Consider

1 **Authorization:** Provide the access token returned by the Token API.

Query Parameters

Name	Description
expectedReceiptId	Use the *ReceiptId* returned from the Expected Receipt API response.

Request Headers

Name	Description
Content-Type	application/json
Authorization	Bearer {token}

Accept	application/json

Request Elements

Name	Data Type	Description
Content Body	CancelERRequest	**Required**. See *Cancel ER Request* below for more details.

Cancel ER Request

Name	Data Type	Description
Company	string	**Required**. Seller company code. Unique code issued by Avectous.
ExpectedReceiptId	long	**Required.** Identifier for the expected receipt to cancel. See *ReceiptId* from Expected Receipt API response.
ReasonCode	string	**Optional.** Reason for cancellation.

Response Elements

Name	Data Type	Description
LogId	Guid	Unique identifier for the request.
Status	string	Processing result for the expected receipt cancellation request. Possible status codes returned: • ***Success*** – expected receipt cancelled. • ***Error*** – expected receipt could not be cancelled.
Messages	Message[]	A list of status messages for errors/warnings. *Note: a successful cancellation will return an empty list.*

Message Object

Name	Data Type	Description
Code	string	Status code for the raised error or warning.
Context	string	Processing step where error or warning was raised.

		Possible values returned: • ***Validate*** – error occurred while validating the cancel expected receipt request. • ***Channel Upload*** – error occurred while sending cancel expected receipt request to the fulfillment channel system.
Severity	string	Level of impact for the raised error or warning. Possible severity levels returned in decreasing order: • ***Fatal*** – critical error occurred leading to unexpected/severe results. • ***Warning*** – event occurred which may result in potential errors.
Message	string	Description of error or warning event that occurred.

Sample Request

```
curl -X PUT https://.../connect/api/v1/expectedreceipt/12345/cancel \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-H "Accept: application/json" \

-d {CancelERRequest Object}
```

Sample Response

```
{
    "LogId": " 443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Success",
    "Messages": []
}
```

Sample Error Response

```
{
    "LogId": "443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Error",
    "Messages": [
        {
            "Code": "PSE",
            "Context": "Pre-Store Validation",
            "Severity": "Fatal",
            "Message": "No Expected Receipt was found"
        }
    ]
}
```

Create a Work Order

HTTP Request

POST https://<client_url>/connect/api/v1/workOrder

Description

To submit a work order.

Things to Consider

1 **Authorization:** Provide the access token returned by the Token API.

Request Headers

Name	Description
Content-Type	application/json
Authorization	Bearer {token}
Accept	application/json

Request Elements

Name	Data Type	Description

Content Body	WorkOrderObject	**Required**. Please see *WorkOrder* JSON file for definitions.

Response Elements

Name	Data Type	Description
LogId	Guid	Unique identifier for the request.
Status	string	Processing result for the work order request. Possible status codes returned: • ***Success*** – work order downloaded. • ***Error*** – work order could not be downloaded.
Messages	Message[]	A list of status messages for errors/warnings. *Note: a successful work order will return an empty list.*

Message Object

Name	Data Type	Description

Code	string	Status code for the raised error or warning.
Context	string	Processing step where error or warning was raised. Possible values returned: • ***Validate*** – error occurred while validating the work order request. • ***Channel Upload*** – error occurred while sending work order request to the fulfillment channel system.
Severity	string	Level of impact for the raised error or warning. Possible severity levels returned in decreasing order: • ***Fatal*** – critical error occurred leading to unexpected/severe results. • ***Warning*** – event occurred which may result in potential errors.
Message	string	Description of error or warning event that occurred.

Sample Request

```
curl -X POST https://.../connect/api/v1/workOrder \
-H "Content-Type: application/json" \
-H "Authorization: Bearer 123456" \
-H "Accept: application/json" \

-d {Work Order Object}
```

Sample Successful Response

```
{
    "LogId": " 443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Success",
    “ReceiptId” : 63,
    "Messages": []
}
```

Sample Error Response

```
{
    "LogId": "443a8639-b1c9-4a92-88fd-64bd20aa4ef5",
    "Status": "Error",
    “ReceiptId” : null,
    "Messages": [
        {
            "Code": " WO006",
            "Context": " Validate ",
            "Severity": "Fatal",
            "Message": " Duplicate WorkOrderNumber TestOrderNumber31 in WorkOrder"
        },
        {
            "Code": " WO008",
            "Context": " Validate ",
            "Severity": "Fatal",
```

```
            "Message": "Missing StagingWorkOrderDetails"
        }
    ]
}
```

Get Product Updates

HTTP Request

GET https://<client_url>/connect/api/v1/product?modifiedStartDate={date}&modifiedEndDate={date}&companyCode={string}

Description

To retrieve information on products updated between specific date ranges for a specific company.

Things to Consider

1. **Authorization:** Provide the access token returned by the Token API.
2. **modifiedStartDate** and **modifiedEndDate** are both UTC Datetime with format yyyy-MM-ddThh:mm:ss
3. The response returns only Product and Product UOM details.

Query Parameters

Name	Data Type	Description
modifiedStartDate	UTC Datetime	**Required.**
modifiedEndDate	UTC Datetime	**Required.**
companyCode	string	**Required.**

Request Headers

Name	Description
Authorization	Bearer {token}

Response Elements

Name	Data Type	Description
Product	Product[]	A list of *Products*. Please see *Product* file for details.

Product Object

Name	Data Type	Description
ProductUOMs	ProductUOM[]	Please see *Product* file > *ProductUOM* tab for details.

Sample Request

```
curl -X GET https://.../connect/api/v1/product?modifiedStartDate=2021-07-
20T19:04:24&modifiedEndDate=2021-07-28T19:04:24&companyCode=Test\
-H "Authorization: Bearer 123456" \
```

Get Inventory Details (Current Snapshot View)

HTTP Request

GET https://<client_url>/connect/api/v1/inventorydetail?companyCode={string}&warehouseCode={string}

Description

To retrieve a current snapshot of inventory by company and warehouse. This replaces the legacy Inventory Snapshot API.

Things to Consider

1 **Authorization:** Provide the access token returned by the Token API.

Query Parameter

Name	Data Type	Description
companyCode	string	**Required.**
warehouseCode	string	**Required.**

Request Headers

Name	Description
Authorization	Bearer {token}

Response Elements

Name	Data Type	Description
InventoryDetails	InventoryDetail[]	A list of *InventoryDetail* items. Please see *InventoryDetail* file for definitions.

Sample Request

```
curl -X GET https://.../connect/api/v1/inventorydetail?companyCode=Test&warehouseCode=1
\
-H "Authorization: Bearer 123456" \
```