

## Commerce Partner API Release Notes

Documentation Version	Release Notes
December 03, 2024 API Doc Version: 2024.12.1	<ul style="list-style-type: none"> <li>Acrobat Pro High Growth Offers for existing customers. These offers will replace the current High Volume Discounting program that exists in VIP. See <a href="#">Manage High Growth Offers</a> for more details.</li> <li>Included customer's VIP renewal status within migration preview API response. See <a href="#">Preview Offer</a> for more details.</li> </ul> <p>Upcoming releases</p> <ul style="list-style-type: none"> <li>HVD customers in VIP can now migrate to VIP Marketplace. See <a href="#">Migrate High Volume Discount customers from VIP to VIP Marketplace</a> for more details.</li> </ul>
November 01, 2024 API Doc Version: 2024.11.1	<ul style="list-style-type: none"> <li>Acrobat Pro High Growth Offers for existing customers. These offers will replace the current High Volume Discounting program that exists in VIP. See <a href="#">Manage High Growth Offers</a> for more details.</li> </ul>
October 18, 2024 API Doc Version: 2024.10.1	<ul style="list-style-type: none"> <li>Ability to place renewal orders after anniversary date <ul style="list-style-type: none"> <li>Added allowedActions field in the <b>Get All Subscriptions for a Customer</b> API to indicate the subscriptions that can be selected for manual renewals. See <a href="#">Get Subscription Details</a> for more information.</li> <li>The newly introduced RENEWAL order type in the Create Order API is used for renewals that are initiated after the anniversary date. See <a href="#">Create Order Scenarios</a> for more details.</li> </ul> </li> <li>Enabling Global customers for Create Subscription <ul style="list-style-type: none"> <li>Global customers can now use Create Subscription API to create scheduled subscriptions. See <a href="#">Create Subscriptions</a> for more details.</li> </ul> </li> </ul>
September 24, 2024 API Doc Version: 2024.9.2	<ul style="list-style-type: none"> <li>Create and manage linked memberships Adding support for creating and managing linked memberships that facilitates combining purchases across linked customer accounts to achieve better volume discounts. See <a href="#">Manage linked memberships</a> for more information.</li> <li>“Late renewals” or “Renewal” Order Type Introducing a new Order Type for partners to place “late renewal” after anniversary date. See <a href="#">Create Order</a> for more information.</li> </ul>
September 17, 2024 API Doc Version: 2024.9.1	<ul style="list-style-type: none"> <li>Create orders <ul style="list-style-type: none"> <li>Implemented a validation to prevent from placing new orders before the subscription’s anniversary date. If a partner attempts</li> </ul> </li> </ul>

	to place an order before this date, an error message will appear as the response.
August 23, 2024 API Doc Version: 2024.8.2	<ul style="list-style-type: none"> <li>Worldwide offers: <ul style="list-style-type: none"> <li>Added ability to enable a customer for worldwide (global) sales</li> <li>Added ability to create “Deployments” for a customer to purchase worldwide offers</li> <li>Added an optional currencyCode to the order lineItem level to support global deployments</li> <li>Added deploymentId to order lineItem (read/write)</li> <li>Added deploymentId and currencyCode to subscription resource (read only)</li> <li>Added deploymentId and currencyCode to transfer resource (read only)</li> </ul> </li> <li>Included 3yc info in preview migration and preview/execution of reseller change API calls.</li> </ul>
August 01 <sup>st</sup> , 2024 API Doc Version: 2024.8	<ul style="list-style-type: none"> <li>Create Subscription: <ul style="list-style-type: none"> <li>Added ability to create future dated subscription that would become active on next anniversary date.</li> </ul> </li> </ul> See <a href="#">Create Subscription</a> for more information.
June 10 <sup>th</sup> , 2024 API Doc Version: 2024.6	<ul style="list-style-type: none"> <li>Added a new API to fetch all customers (under a given reseller) who have added seats via Adobe’s Admin console. See <a href="#">Get seat addition Notification</a> for more information.</li> <li>Added a new API to fetch all seats/licenses added by a given customer via admin console. See <a href="#">Get seat addition details for Customer</a> for more information.</li> <li>These APIs will inform our partners to place corresponding orders backing the seats added by the customers via admin console.</li> <li>Added new Error Code for Preview / Create Transfer response. See <a href="#">Status Codes &amp; Error Handling</a> for more information.</li> </ul>
February 29, 2024	<ul style="list-style-type: none"> <li>Added self-service capability for a customer and partner to move between resellers/partners. See <a href="#">Reseller Change Process</a> for more information.</li> </ul>

In the table of contents and throughout this document, sections that have been updated for this release are **highlighted in yellow**.

<b>Introduction</b>	<b>5</b>
<b>Identity Management &amp; Authentication</b>	<b>6</b>
<b>Health Check</b>	<b>7</b>
1. Ping	7
2. Authenticated Ping	7
<b>Reseller Account Management</b>	<b>8</b>
3. Create Reseller Account	8
4. Get Reseller Account	11
5. Update Reseller Account	12
<b>Customer Account Management</b>	<b>12</b>
<b>Market Segment</b>	<b>13</b>
Market Segment API Changes	13
Market Segment Workflows	14
<b>Three Year Commit</b>	<b>15</b>
Introduction to Three Year Commit	15
3YC Workflows	15
3YC API Changes	16
3YC Flow Diagrams	19
<b>Manage linked memberships</b>	<b>19</b>
How to obtain more discounts through linked membership?	20
Creating new linked memberships using API	21
Get details of linked membership	23
Update the linked membership to add members	24
Migrate linked membership from VIP to VIP MP	25
<b>Manage High Growth Offers</b>	<b>25</b>
Who is eligible for High Growth Offers?	25
High Growth Offer workflows	26
High Growth Offer scenarios	27
Manage High Growth Offers through APIs	29
<b>Migrate High Volume Discount customers from VIP to VIP Marketplace</b>	<b>38</b>
Migration workflow	39
HVD migration using API	40
6. Create Customer Account	44
<b>7. Get Customer Account</b>	<b>48</b>
8. Update Customer Account	51
9. Get Licenses Pending Partner Order details for Customer	53
<b>Deployment Management</b>	<b>54</b>
10. Create Deployment	54
11. Get Deployment	56
12. Update Deployment	57
13. Get Deployments for Customer	59
<b>Reseller Change Process</b>	<b>60</b>

Reseller Change Process API	63
14. Preview Transfer	63
15. Commit Transfer	65
16. Get Transfer	67
<b>Order Management</b>	<b>68</b>
Volume Discounting	68
17. Create Order	69
Create Order Scenarios	71
18. Get Order Details	78
19. Update Order ExternalReferenceId	79
20. Get Order History for Customer	80
<b>Subscription Management</b>	<b>82</b>
Auto-Renewal Scenarios	82
21. Get Subscription Details	83
22. Get All Subscriptions for a Customer	85
23. Update Subscription Auto-Renewal	87
24. Create Subscription	88
<b>VIP to VIPMP Seamless Move (Transfers):</b>	<b>91</b>
Seamless Move Eligibility Rules	91
25. Preview Offers	92
26. Transfer Subscriptions	95
27. Get Transfer Details	97
<b>Notification Management:</b>	<b>98</b>
28. GET LICENSES_PENDING_PARTNER_ORDER Notification	99
<b>Appendix</b>	<b>100</b>
29. Headers	100
30. Idempotency – Correlation ID Header	100
31. HTTP Status Codes	101
32. Status Codes & Error Handling	102
33. Supported Countries and Locales	117
34. Resources and Fields	120
35. Validations/Regular Expressions	127
36. Support	128

## Introduction

This API allows partner cloud marketplaces to create and manage both Reseller and Customer accounts, as well as place orders of select Adobe products for those Customers. Before placing an Order, both a Reseller and Customer account must exist. Each Customer must be tied to a Reseller, and each Reseller must be tied to a Distributor.

The Distributor → Reseller → Customer → Order relationships are unidirectionally one-to-many. This means one Distributor can have many Resellers, but each reseller may only belong to one Distributor. Similarly, a Reseller may have many Customers, but a Customer can only be tied to one Reseller. If a reseller wishes to work with multiple distributors, a new Reseller account must be created for each Distributor. This is the same for customers with multiple resellers: a new Customer account must be created for each Reseller.

- Marketplace presents Adobe Terms & Conditions to Reseller
- If a Reseller accepts T&C, marketplace creates an account for reseller with **Create Reseller Account**
- Sometime prior to an order being placed, Reseller creates an account for a Customer and the marketplace calls **Create Customer Account**
  - Customer resource is returned with link to **Get Customer Account Details**
  - The end customer's address and contact information can be updated through a call to **Update Customer Account**
- Once a customer account exists, the Reseller places an order in the marketplace. Marketplace then calls **Create Order** for all eligible Adobe products in the order
  - Order resource is returned with links to **Get Order Details** for the order itself and **Get Subscription Details** for any subscriptions associated with the order
  - Orders can be cancelled within 14 days of placing the order. When a Reseller cancels an order in the marketplace, the marketplace should call **Cancel Order** to send a cancellation to Adobe
- Subscriptions that are created by an order will be set to auto-renew by default. This option, along with the quantity of licenses to renew, can be updated using an **Update Subscription Auto-Renewal** call

Customers can also be transferred from the VIP buying program to VIPMP. This is done through the transfer APIs (Preview Offers, Transfer Subscriptions, and Get Transfer Details). The **Preview Offers** call is an optional API to preview a customer's eligibility for transfer, as well as their current subscriptions and renewal date(s).

## Identity Management & Authentication

The Commerce Partner API uses Adobe's standard JWT (Service Account) Authentication.

All API calls for a distributor originate from that distributor's IMS Organization. After a direct partner signs the contract and provides necessary onboarding information, Adobe creates an IMS Organization for that partner with admins that the partner specified. Admins can manage users (other admins and developers) using the [Adobe Admin Console](#). Once the organization is created, production credentials can be generated self-service through the [Adobe Developer Console](#). System Admins and Developers of the organization will have access to the Developer Console.

All API calls (with the exception of /ping) require a valid IMS access token. To get a valid access token, you must use the credentials (from the developer console) to generate a JWT and exchange the JWT for an access token using the IMS exchange API. For more information on creating a project in the Developer Console and retrieving an access token, please visit these links:

- <https://www.adobe.io/authentication/auth-methods.html#!AdobeDocs/adobeio-auth/master/AuthenticationOverview/ServiceAccountIntegration.md>
- <https://www.adobe.io/authentication/auth-methods.html#!AdobeDocs/adobeio-auth/master/JWT/JWT.md>

For access to the sandbox environment, credentials will be created by Adobe and shared. The authentication flow is the same in the sandbox environment (generating JWT, exchanging JWT for access token) – the difference is in the Partner API URL and the IMS URL.

	Sandbox	Production
<b>Partner API Base URL</b>	<a href="https://partnersandbox-stage.adobe.io">https://partnersandbox-stage.adobe.io</a>	<a href="https://partners.adobe.io">https://partners.adobe.io</a>
<b>IMS Base URL</b>	<a href="https://ims-na1-stg1.adobelogin.com">https://ims-na1-stg1.adobelogin.com</a>	<a href="https://ims-na1.adobelogin.com">https://ims-na1.adobelogin.com</a>
<b>Metascope</b>	ent_partners_sdk	ent_partners_sdk

## Health Check

### 1. Ping

**GET** /ping

**REQUEST HEADERS:**

**X-Api-Key:** <your-api-key>

**REQUEST BODY:** None

**RESPONSE BODY:**

pong

### 2. Authenticated Ping

**GET** /partnerservice/ping

**REQUEST HEADERS:**

**X-Api-Key:** <your-api-key>

**Authorization:** Bearer <token>

**REQUEST BODY:** None

**RESPONSE BODY:**

pong

**Assumptions:**

- The authenticated ping endpoint is used to verify that the service is available and that your API Key and Authorization Token are valid

## **Reseller Account Management**

### **3. Create Reseller Account**

**POST** /v3/resellers

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** Reseller resource without read-only fields

```
{
  "distributorId" : "345434543",
  "externalReferenceId" : "888",
  "companyProfile" : {
    "companyName" : "Fairmont",
    "preferredLanguage" : "en-US",
    "marketSegments" : ["COM", "EDU"],
    "address" : {
      "country" : "US",
      "region" : "CA",
      "city" : "San Jose",
      "addressLine1" : "200 Fairmont Ave",
      "addressLine2" : "Apt 123",
      "postalCode" : "95110-1234",
      "phoneNumber" : "800-123-4567"
    },
    "contacts" : [{
      "firstName" : "Mickey",
      "lastName" : "Mouse",
      "email" : "mickey@mouse.com",
      "phoneNumber" : "408-123-4567"
    }]
  }
}
```



**RESPONSE BODY:** Reseller resource

```
{
  "distributorId": "345434543",
  "externalReferenceId": "888",
  "resellerId": "5556667778",
  "companyProfile": {
    "companyName": "Fairmont",
    "preferredLanguage": "en-US",
    "marketSegments": ["COM", "EDU"],
    "address": {
      "country": "US",
      "region": "CA",
      "city": "San Jose",
      "addressLine1": "200 Fairmont Ave",
      "addressLine2": "Apt 123",
      "postalCode": "95110-1234",
      "phoneNumber": "800-123-4567"
    },
    "contacts": [{
      "firstName": "Mickey",
      "lastName": "Mouse",
      "email": "mickey@mouse.com",
      "phoneNumber": "408-123-4567"
    }]
  },
  "creationDate": "2019-05-02T22:49:52Z",
  "status": "1000",
  "links": {
    "self": {
      "uri": "/v3/resellers/5556667778",
      "method": "GET",
      "headers": []
    }
  }
}
```

**HTTP STATUS CODES:**

201	Reseller account successfully created
400	Bad request
401	Invalid Authorization token
403	Invalid API Key

## Assumptions:

- Partner will present reseller with T&Cs and make this call after reseller agrees
- Reseller Account ID is created by this service and returned synchronously
- Market segment is only commercial for M1
- externalReferenceId should be used to pass in the marketplace's identifier for the reseller
  - Optional and does not need to be unique
- Distributor ID -> Reseller ID -> Customer ID relationship must be unique

- Synchronous call will return an account with status "1002" (Pending) while final validation happens asynchronously. The account may then become Active or Inactive. Pending resellers can create customers, but not update the reseller account.
- Contact names for existing email addresses cannot be changed through this API. If you send a different name for an existing email, that name will be returned in the synchronous response, but it will not persist in our system. The existing first and last name of the contact will be used and returned in any GET calls
- Resellers can be created with optional market segments
  - If no market segments are sent, the reseller will be created with COM
- Resellers may sell into multiple market segments but must always have at least one (default COM)

4. Get Reseller Account

**GET** /v3/resellers/<reseller-id>

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** None

**RESPONSE BODY:** Same as Create Reseller (Reseller resource)

```
{
  "distributorId": "345434543",
  "externalReferenceId": "888",
  "resellerId": "5556667778",
  "companyProfile": {
    "companyName": "Fairmont",
    "preferredLanguage": "en-US",
    "marketSegments": ["COM", "EDU"],
    "address": {
      "country": "US",
      "region": "CA",
      "city": "San Jose",
      "addressLine1": "200 Fairmont Ave",
      "addressLine2": "Apt 123",
      "postalCode": "95110-1234",
      "phoneNumber": "800-123-4567"
    },
    "contacts": [{
      "firstName": "Mickey",
      "lastName": "Mouse",
      "email": "mickey@mouse.com",
      "phoneNumber": "408-123-4567"
    }]
  },
  "creationDate": "2019-05-02T22:49:52Z",
  "status": "1000",
  "links": {
    "self": {
      "uri": "/v3/resellers/5556667778",
      "method": "GET",
      "headers": []
    }
  }
}
```

**HTTP STATUS CODES:**

200	Reseller account details successfully returned
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid reseller ID

5. Update Reseller Account

**PATCH** /v3/resellers/<reseller-id>

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** CompanyProfile for reseller with optional externalReferenceId

```
{
  "externalReferenceId" : "999",
  "companyProfile" : {
    "companyName" : "Fairmont",
    "preferredLanguage" : "en-US",
    "marketSegments" : ["COM", "EDU"],
    "address" : {
      "country" : "US",
      "region" : "CA",
      "city" : "San Jose",
      "addressLine1" : "200 Fairmont Ave",
      "addressLine2" : "Apt 123",
      "postalCode" : "95110-1234",
      "phoneNumber" : "800-123-4567"
    },
    "contacts" : [
      {
        "firstName" : "Mickey",
        "lastName" : "Mouse",
        "email" : "mickey@mouse.com",
        "phoneNumber" : "408-123-4567"
      }
    ]
  }
}
```

**RESPONSE BODY:** Reseller resource

```
{ Reseller resource }
```

**HTTP STATUS CODES:**

200	Reseller account successfully updated
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Reseller ID

Assumptions:

- Update replaces existing data
- Non-editable fields (in red) need to be passed in but need to match the original values. Otherwise, an error is returned
- externalReferenceId can now be updated

**Customer Account Management**

Market Segment

Market Segment API Changes

Reseller Resource

- **Market Segments:** New field *companyProfile.marketSegments*
  - List of market segments the reseller can sell into
  - Can be set during Create Reseller and Update Reseller
  - Default is [COM]

Customer Resource

- **Market Segment:** New field *companyProfile.marketSegment*
  - Can be set during Create Customer and Update Customer
  - A customer may only have a single market segment
  - Default is COM
- **Market Sub-Segments:** New field *companyProfile.marketSubSegments*
  - Read Only: Market subsegments are set by the customer in the Adobe Admin Console
  - Currently subsegments are only applicable for the EDU market segment

Market Segment (reseller/customer)

value	description
COM	Commercial (default)
EDU	Education
GOV	Government

Market Sub-Segment (customer only)

value	description	offers available to purchase
K_12	K-12 School (Primary or Secondary School)	<ul style="list-style-type: none"><li>• EDU offers without a sub-segment</li><li>• EDU offers with K-12 sub-segment</li></ul>
HIGHER_ED	Higher Education Institution	<ul style="list-style-type: none"><li>• EDU offers without a sub-segment</li><li>• EDU offers with HED sub-segment</li></ul>
NON_PROFIT	Nonprofit Organization	<ul style="list-style-type: none"><li>• EDU offers without a sub-segment</li></ul>

- Note:** The *Additional Detail* (Column P) in the price list file contains information on the sub-segment of an offer (if applicable) as well as the minimum purchase quantity (if applicable)
- If the offer has a sub-segment, the *Additional Detail* text will **begin** with that subsegment (*K-12* or *HED*)
  - If the offer has a minimum purchase quantity, the *Additional Detail* text will **end** with that information

Example Additional Detail: K-12 Shared Device Site Education License Lab and Classroom (100+)

- This offer is associated with the K-12 sub-segment and has a minimum purchase quantity of 100

1. **Setting Market Segments (Reseller)**

- Reseller market segments can be set as part of a create or update request
- A reseller's market segments may be changed at any point, but removing a market segment may cause future orders/renewals to fail
- All existing resellers with no market segment will be initialized with only COM market segment

[\[POST Create Reseller API\]](#)

[\[PATCH Update Reseller API\]](#)

2. **Setting Market Segment (Customer)**

- Customer market segment can be set as part of a create or update request
- A customer's market segments can only be changed if the customer has no active subscriptions
- All existing customers with no market segment will be initialized to COM

[\[POST Create Customer API\]](#)

[\[PATCH Update Customer API\]](#)

3. **Setting Market Sub-Segment (Customer only)**

- A customer's market sub-segment can only be set by the customer in the Adobe Admin Console
- SANDBOX: For testing, the sub-segment can be set in the sandbox partner portal

[\[GET Customer API\]](#)

4. **Transferring Customers to your marketplace (Customer only)**

- Customers with any market segment can be transferred as long as the new reseller is enabled to sell into the customer's market segment

[\[POST Create Transfer API\]](#)

[\[GET Customer API\]](#)

5. **Purchasing Offers for a Market Segment/Subsegment**

- Customers may only purchase offers in their market segment/subsegment
- Resellers may only purchase offers in a market segment they are enabled for
- Requests to purchase offers in a different segment will return an error (2129 – Reason Code: INELIGIBLE\_MARKET\_SEGMENT)

[\[POST Create/Preview Order API\]](#)

**New/Updated Error Codes**

Error Code	Error Message	Applicable APIs	Change Details
2135 (new error code)	Invalid market segment used for customer	Create Customer, Update Customer	New Error Code
2129 (existing error code)	Customer is not eligible to purchase Offer ID	Preview Order, Create Order	New Additional Details Reason Code: INELIGIBLE_MARKET_SEGMENT

## Three Year Commit

### Introduction to Three Year Commit

3-Year Commit (3YC) is a loyalty program that allows a customer to get a **greater discount level** as well as a **price lock** for 3 terms (the current term and 2 additional terms). The customer commits to a minimum quantity that they must purchase *and maintain* throughout the 3-year commitment period. The minimum commit quantities are as follows:

- **Minimum commit quantity for licenses:** 10
- **Minimum commit quantity for consumables:** 1000

It is important to use the Preview Order API to determine the correct discount level. Preview Order uses the customer's 3YC commitment as well as any accepted commitment requests in order to return the Offer ID with the best discount.

### 3YC Workflows

There are four ways that a customer can become enrolled in 3YC. In all scenarios, the customer is required to accept the commitment terms (the minimum quantities and commitment end date) in the Adobe Admin Console before receiving any discounts.

1. **New Customer (partner-initiated):** During customer creation, partners may set the requested minimum quantities, which would start the 3YC customer acceptance workflow after the account becomes active.  
[\[POST Create Customer API\]](#)  
[\[GET Get Customer API\]](#)
2. **Existing Customer (partner-initiated):** Existing customers (with active 3YC or not) can be updated with new requested minimum quantities, which would start the 3YC customer acceptance workflow once the account is updated from the request.  
[\[PATCH Update Customer API\]](#)  
[\[GET Get Customer API\]](#)
3. **Existing Customer (customer-initiated):** Existing customers (not already in 3YC) that meet the 3YC requirements may be presented an offer in Adobe admin Console to enter 3YC *with the minimum quantities set to the customer's current quantities*.  
[\[GET Get Customer API\]](#)
4. **Existing VIP Customer (partner-initiated):** Existing VIP customers with active 3YC can be transferred to VIPMP and retain their 3YC commitment terms, which can be retrieved using the GET Customer API.  
[\[POST Transfer Subscriptions API\]](#)  
[\[GET Get Customer API\]](#)

## 3YC API Changes

### Customer Resource

- New *benefits* array of *benefit* objects
- New potential 3YC discount levels
  - **LICENSE:** 12, 13, 14
  - **CONSUMABLES:** TB, TC, TD, TE, TF, TG
- Discount levels will be updated whenever the commitment status changes (new commitment or expiration)
- During Order Preview, if the customer has an **ACCEPTED** *commitmentRequest* for a quantity that would be reached in that order, the preview response will use the discount level for that *commitmentRequest*
  - Placing the order for that quantity would make the *commitmentRequest* **COMMITTED** as well as creating the *commitment* object
- Customer can be enrolled in 3YC for either **LICENSE** or **CONSUMABLES** or both.
  - If only enrolled for one, there is no minimum quantity for the other type and the customer gets no discount for that type
  - Can enroll in one *offerType* first and then add the other later. The committed *offerType* as well as the new *offerType* must be included in *commitmentRequest*



Object	Sample Request	Notes
<b>commitmentRequest</b>	<pre> REQUEST {   "companyProfile":{     "companyName":"Fairmont",     "preferredLanguage":"en-US",     "address":{       "country":"US",       "region":"CA",       "city":"San Jose",       "addressLine1":"200 Fairmont Ave",       "addressLine2":"Apt 123",       "postalCode":"95110-1234",       "phoneNumber":"800-123-4567"     },     "contacts":[      ]   },   "benefits":[     {       "type":"THREE_YEAR_COMMIT",       "commitmentRequest":{         "minimumQuantities":[           {             "offerType":"LICENSE",             "quantity":10           },           {             "offerType":"CONSUMABLES",             "quantity":1000           }         ]       }     }   ] } </pre>	<ul style="list-style-type: none"> <li>Used to request 3YC for a customer without a commitment</li> <li>Used to request 3YC quantity increase for a customer with an existing commitment</li> <li>Will overwrite existing <i>commitmentRequest</i></li> <li>Can not be requested alongside <i>recommitmentRequest</i> or if customer has an existing <i>recommitmentRequest</i></li> <li>Does not need to include quantities for all offerTypes</li> <li>endDate will be 2 years after the current cotermdate             <ul style="list-style-type: none"> <li>If customer does not have a cotermdate (they have not placed an order) at the time of acceptance, the endDate will be 3 years from the acceptance date and the cotermdate will be set at that point</li> </ul> </li> </ul>

recommitmentRequest

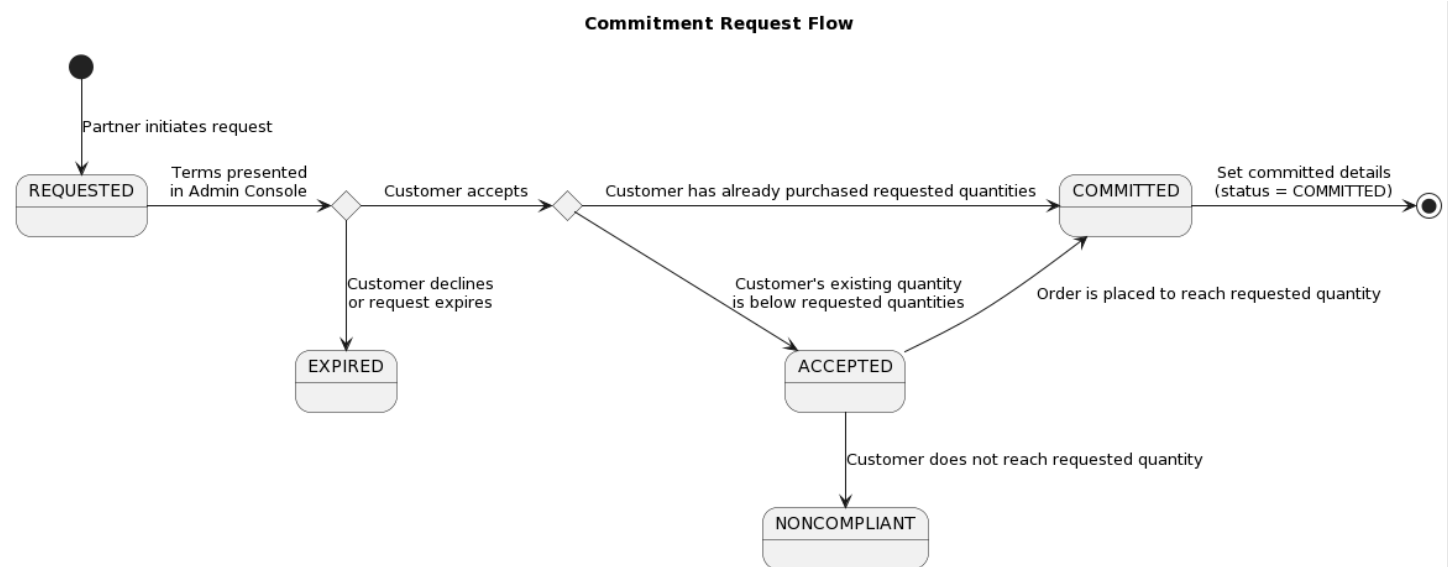
```
REQUEST
{
  "companyProfile":{
    "companyName":"Fairmont",
    "preferredLanguage":"en-US",
    "address":{
      "country":"US",
      "region":"CA",
      "city":"San Jose",
      "addressLine1":"200 Fairmont Ave",
      "addressLine2":"Apt 123",
      "postalCode":"95110-1234",
      "phoneNumber":"800-123-4567"
    },
    "contacts":[

    ]
  },
  "benefits":[
    {
      "type":"THREE_YEAR_COMMIT",
      "recommitmentRequest":{
        "minimumQuantities":[
          {
            "offerType":"LICENSE",
            "quantity":10
          },
          {
            "offerType":"CONSUMABLES",
            "quantity":1000
          }
        ]
      }
    }
  ]
}
```

- Used to request 3YC recommitment
- Must have existing commitment and less than 30 days before the commitment end date
- Will overwrite existing recommitmentRequest
- Can not be requested alongside *commitmentRequest* or if customer has an existing *commitmentRequest*

## 3YC Flow Diagrams

The diagram below shows the lifecycle of the commitmentRequest object. The same lifecycle applies for the recommitmentRequest, with the difference being that recommitmentRequest can't move to COMMITTED status until the current commitment ends.



```

"benefits":[
  {
    "type":"THREE_YEAR_COMMIT",
    "commitment":{...},
    "commitmentRequest":{
      "startDate":"2022-07-17",
      "endDate":"2025-04-28",
      "status":"REQUESTED",
      "minimumQuantities":[
        {
          "offerType":"LICENSE",
          "quantity":10
        },
        {
          "offerType":"CONSUMABLES",
          "quantity":1000
        }
      ]
    },
    "recommitmentRequest":{...}
  }
]

```

## Manage linked memberships

Linked Membership is an optional feature of the Adobe Value Incentive Plan Marketplace (VIP MP). This plan allows organizations affiliated with VIP MP to combine purchases across linked customer accounts to achieve [better volume discount levels](#), while still managing and purchasing licenses separately.

You can provide linked memberships to customers from the following types of organizations:

- Commercial affiliates
- Government agencies
- Primary and secondary education institutions, such as school districts and state schools
- Departments with separate budgets and license administration needs

There are two types of Linked Memberships:

- **Standard:** Suitable for departments or sub-organizations with separate budget or administrative requirements, including business affiliates, school districts, state schools, government departments, and similar entities.
- **Consortiums:** An association or combination of organizations with similar interests and objectives, managed by a controlling entity.

How to obtain more discounts through linked membership?

The linked membership feature helps organizations to achieve better volume discounts, which are determined based on the number of licenses purchased during a license term. Organizations purchasing 10 or more licenses qualify for volume discounts. As the number of licenses increases, the discount level rises accordingly, as illustrated below:



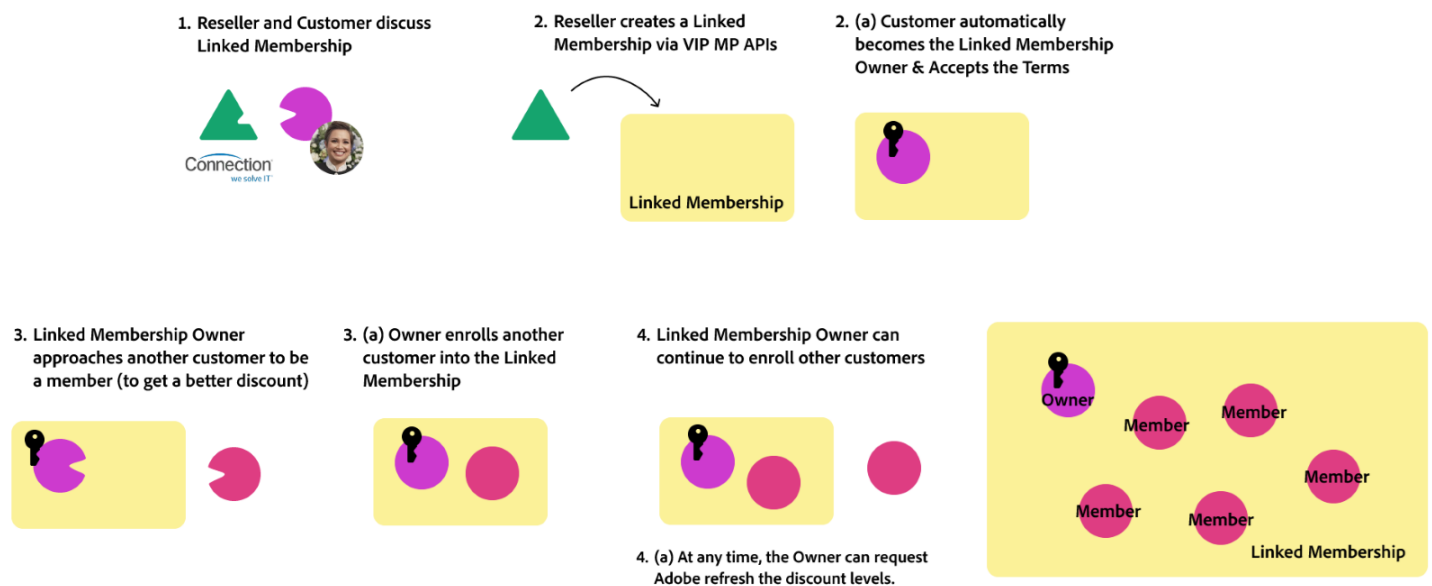
With linked memberships, organizations affiliated with VIP MP can combine or aggregate their licenses or transactions with their affiliates to secure better discounts. For example, *Organization A* has 70 licenses and falls into Discount Level 3, as indicated in the previous table. *Organization B* has 31 licenses and gets 5% discount defined for Level 2. By combining these organizations through linked membership, now they have 101 licenses and qualify for level 4 (15%) discounts.

You can include offers for all volume discounted price levels and scaled pricing models. The following table lists the discount tiers for Sign Enterprise transactions:

Transaction range	Tier
1 to 999	T1
1000 to 2499	T2

2500 to 4999	T3
5,000-14,999	T4
15,000 to 49,999	T5
50,000 to 99,999	T6
> 100K	T7

Resellers can create new linked memberships using the [Update Customer Account API](#). Each linked membership has one owner and multiple members. The owner of a linked membership and its members are B2B customer organizations. The owner must agree to the [terms and conditions](#) in the Admin Console and enroll other members into the linked membership. The workflow for creating a linked membership and adding members is as follows:



The following sections provide how to manage different stages of this workflow:

- [Create new linked memberships](#)
- [Get details of linked memberships](#)
- [Update a linked membership to add members](#)

## Creating new linked memberships using API

The following points highlight key aspects of linked membership creation:

- Linked membership account details are maintained independently of the VIP MP customer account details.
- Discount levels for linked memberships are automatically calculated on the anniversary date of the linked membership. The calculated discount level is then applied to all members and the owner.
- The calculated discount level is propagated to all the participating linked members and the owner.
- A linked membership owner and members are not eligible for the three-year commit benefits.

Partner marketplaces can create linked memberships using the [Update Customer API](#).

### Request

The following example shows the parameters to be used in the request body of the [Update Customer API](#) endpoint to define a new linked membership:

Adobe Confidential Information – Do Not Forward

```

"linkedMembership": {
  "type": "STANDARD",
  "name": "This is the Group Created for 1005516130"
},
"companyProfile": {
  "companyName": "Test Customer 33a0e0bc-e38f-4b65-a5a2-8ccad6ea0e8",
  "preferredLanguage": "en-US",
  "marketSegment": "COM",
  "marketSubSegments": [],
  "address": {
    "country": "US",
    "region": "CA",
    "city": "San Jose",
    "addressLine1": "345 Park Ave",
    "addressLine2": "",
    "postalCode": "95110",
    "phoneNumber": ""
  },
  "contacts": [
    {
      "firstName": "David",
      "lastName": "John",
      "email": "dave.j+LM-3YC-23@adobetest.com"
    }
  ]
}

```

You can use the following CURL command as a reference:

```

curl --location --request PATCH 'https://partners.adobe.io/v3/customers/1005513636' \
--header 'x-api-key: <your-api-id>' \
--header 'X-Request-Id: 2950f0f0-300993c3c261b9a' \
--header 'X-Correlation-Id: 9f5cb46f-59e0-44f2' \
--header 'Content-Type: application/json' \
--header 'Accept: application/json' \
--header 'Authorization: Bearer XXXXX' \
--data-raw '{
  "linkedMembership": {
    "type": "STANDARD",
    "name": "This is the Group Created for 1005516130"
  },
  "companyProfile": {
    "companyName": "Test Customer 33a0e0bc-e38f-4b65-a5a2-8ccad6ea0e8",
    "preferredLanguage": "en-US",
    "marketSegment": "COM",
    "marketSubSegments": [],
    "address": {
      "country": "US",
      "region": "CA",
      "city": "San Jose",
      "addressLine1": "345 Park Ave",
      "addressLine2": "",
      "postalCode": "95110",
      "phoneNumber": ""
    }
  }
}'

```

```
    },
    "contacts": [
      {
        "firstName": "David",
        "lastName": "John",
        "email": "dave.j+LM-3YC-23@adobetest.com"
      }
    ]
  }
}
```

#### Response body

Same as [Get Customer Account Details \(Customer resource\)](#).

**Note:** Any contacts specified in this call will receive the admin welcome email. This can be resent if an end-user did not receive it.

#### HTTP status codes

Status code	Description
200	Account successfully updated
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid customer ID

### Get details of linked membership

The [Get Customer API](#) returns details about linked memberships associated with a customer (owner/member). The API returns the subscription details, including the linked membership details. The following response sample shows details fetched for a customer with a linked membership:

```
{
  "externalReferenceId": "240715235148492799",
  "customerId": "1005512199",
  "resellerId": "1000250355",
  "status": "1000",
  "companyProfile": {
    "companyName": "Test organization for linked membership",
    "preferredLanguage": "en-US",
    "marketSegment": "EDU",
    "marketSubSegments": [],
    "address": {
      "country": "US",
      "region": "CA",
      "city": "SAN JOSE",
      "addressLine1": "3453 Park Ave",
      "addressLine2": "3453 Park Ave",
      "postalCode": "95110",
      "phoneNumber": ""
    }
  },
  "contacts": [
```

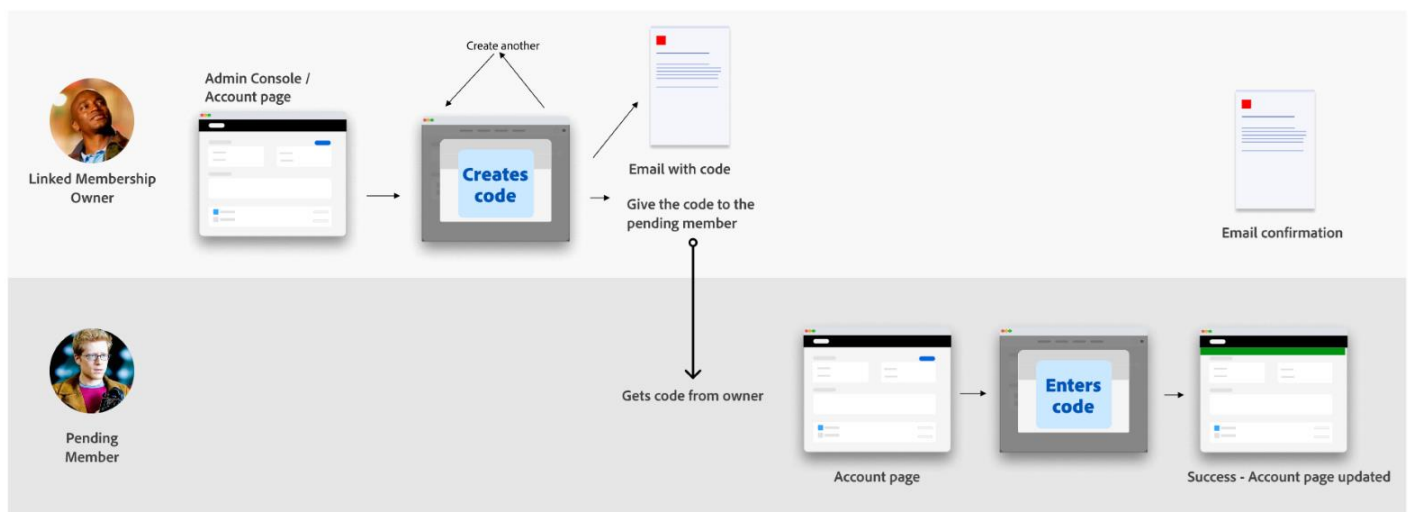
```

    {
      "firstName": "ABC",
      "lastName": "Corporation",
      "email": "email@domain.com",
      "phoneNumber": "28880-7323-8195"
    }
  ],
  "discounts": [
    {
      "offerType": "LICENSE",
      "level": "01"
    }
  ],
  "cotermDate": "2025-07-16",
  "creationDate": "2024-07-15T23:51:49Z",
  "benefits": [],
  "globalSalesEnabled": false,
  "linkedMembership": {
    "id": "51001315",
    "name": "This is the Group Created for 1005513636",
    "type": "STANDARD",
    "linkedMembershipType": "OWNER",
    "creationDate": "2024-07-17T03:47:35"
  },
  "links": {
    "self": {
      "uri": "/v3/customers/1005512199",
      "method": "GET",
      "headers": []
    }
  }
}

```

## Update the linked membership to add members

The workflow for adding new members to the linked membership is illustrated in the following figure:



The Admin Console facilitates the following:



- Enable a linked membership owner to:
  - Accept the terms (Standard/Consortium) within the Admin Console.
  - Generate an authorization code to add a linked member and share the details with the member.
- Facilitate a VIP customer to be a linked membership member.
- Enroll in a linked membership by entering an authorization code provided by a linked membership owner.

Read more about [how to manage linked memberships in the Admin Console](#).

## Migrate linked membership from VIP to VIP MP

- Linked membership account information is displayed during the [VIP Marketplace migration preview](#) API call.
- The migration process requires the linked membership owner to migrate their account before any member migration can occur.

**Note:** Three-year commit (3YC) and linked memberships are mutually exclusive. Plan accordingly when you migrate from VIP to VIP MP.

## Manage High Growth Offers

High Growth Offers are designed for customers who are significantly expanding their existing Acrobat Pro seats. These offers will replace the current High Volume Discounting program (17-23 levels) that exists in VIP.

High Growth Offers are based on the Minimum Order Quantity (MOQ), which defines the purchase quantity and the corresponding discount rates. Any subsequent purchases will adhere to the customer's high growth MOQ offer price.

The available High Growth Offers based on MOQ are:

- MOQ 100
- MOQ 250
- MOQ 500

**Note:** In this document, the terms "High Growth Offer" and "MOQ offer" are used interchangeably.

## Who is eligible for High Growth Offers?

Enterprise and Team customers of Acrobat Pro in the commerce and government market segments can take advantage of High Growth Offers if they meet the following criteria:

1. **Enrollment:** Customers must be enrolled in the [Three-Year Commit \(3YC\) program](#).
2. **MOQ requirement:** Customers require a minimum order quantity (MOQ) of 100 or more.
  - a. Customers can select from the available MOQ offers based on their seat requirements. For example, a customer with 110 seats can opt for either the 250 or 500 MOQ offer option, as shown in the following table.
  - b. MOQ offers are available to both new and existing customers with a certain number of licenses for specific product groups. For example, Acrobat Standard and Acrobat Pro are part of the Acrobat group, and customers with a combined quantity that falls into the range mentioned in the following table become eligible for the corresponding MOQ offers.

Current seats at renewal Acrobat Pro + Standard	Eligible Minimum Order Quantity (MOQ)
≤ 50 seats	100
	250
	500
≤ 125 seats	250
	500

≤ 250 seats	500
-------------	-----

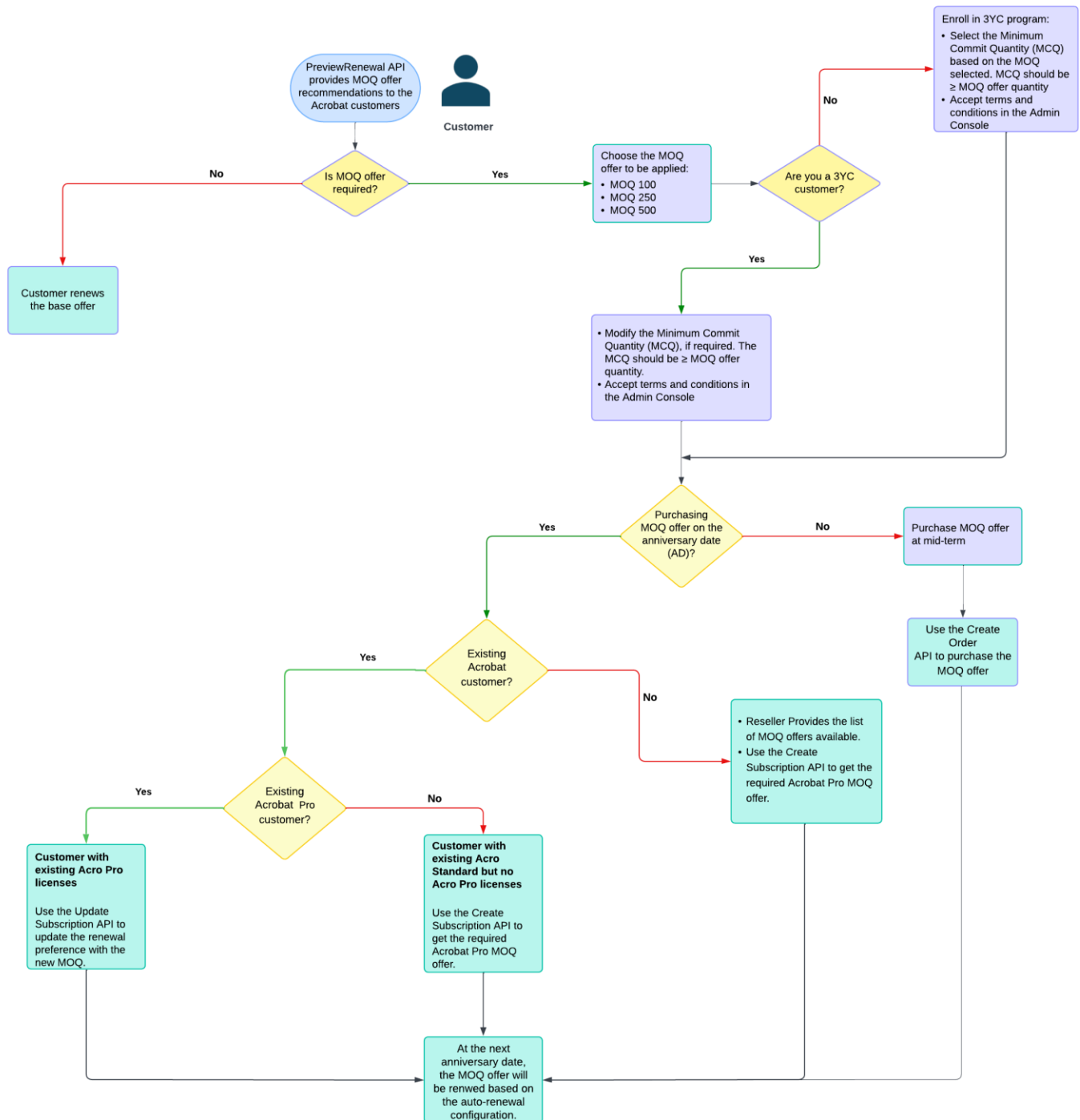
3. **MCQ requirement:** The number of seats committed (MCQ) must be greater than or equal to the number of subscriptions at the renewal date.

Current seats at renewal Acrobat Pro + Standard	Eligible Minimum Order Quantity (MOQ)	3YC Minimum Commit Quantities (MCQ) - Needs to be increased based on the MOQ
≤ 50 seats	100 250 500	100+
≤ 125 seats	250 500	250+
≤ 250 seats	500	500+

High Growth Offer workflows

Customers can opt for an MOQ offer either at the anniversary or mid-term. Purchases made on the anniversary date apply to the next term, while mid-term purchases apply to the current term. All MOQ offer purchases on the anniversary date must be completed within the purchase window from AD-7 to AD-3.

The APIs required to obtain MOQ offers differ depending on various scenarios. The High Growth Offer workflow for these scenarios and the corresponding APIs are shown in the following figure:



## High Growth Offer scenarios

Customers can add licenses or seats mid-term or at the anniversary date, benefiting from the same discount rate as their original MOQ.

### Example scenario

- Initial Purchase: MOQ 100
- Additional Licenses: 300 (Total becomes 400)
- Discount: Same as MOQ 100

Adobe defines the High Growth Offers, and the corresponding offer ID is shared with the partners. For example:

- 65304479CA14X12 = Acrobat Pro 100 MOQ
- 65304479CA14Y12 = Acrobat Pro 250 MOQ
- 65304479CA14Z12 = Acrobat Pro 500 MOQ

You can use this documentation as a reference while finalizing the offer for the customers.

### Hints:

- **AD (Anniversary Date):** The date on which 3YC contracts are renewed.
- **Current quantity:** The number of subscriptions owned by the customer.
- **Renewal quantity:** The number of subscriptions the customer wants to renew on the next anniversary date. This is specified in the auto-renewal configuration.

The following list gives an overview of possible scenarios. You may encounter other scenarios as well. In case of any queries, contact [partnerapi-support@adobe.com](mailto:partnerapi-support@adobe.com).

#### Scenario 1: Normal renewal of subscription at the anniversary date

Example:

Before AD	At AD	Next AD
<ul style="list-style-type: none"> <li>• Quantity of Acrobat Pro: 5</li> <li>• Opted for: MOQ 100 using Update Subscription API</li> </ul>	MOQ 100 of Acro Pro will get renewed	MOQ 100 of Acro Pro will get renewed until the 3YC term ends.

#### Scenario 2: Customer adds Acro Pro seats after opting for MOQ 100

Example:

Before AD	At AD - 7	AD - 3	At AD	At next AD
<ul style="list-style-type: none"> <li>• Quantity of Acrobat Pro: 5</li> </ul>	Opted for MOQ 100 by using the Update Subscription API with renewalQuantity 100.	The customer added 50 Acrobat Pro base offer seats. The current quantity becomes 55.	MOQ 100 of Acro Pro will get renewed.  The customer will still receive the MOQ offer and will not be penalized for adding Acro Pro seats after opting for MOQ.	MOQ 100 of Acro Pro will get renewed until the 3YC term ends.

#### Scenario 3: Customer does not opt for Acrobat Pro MOQ

Example:

Before AD	At AD
<ul style="list-style-type: none"> <li>• Quantity of Acrobat Pro: 120</li> <li>• Renewal quantity: 30</li> </ul>	The customer does not opt for MOQ offers.  The base offer of Acrobat Pro will get renewed with the renewal quantity 30.

#### Scenario 4: Customer purchases MOQ offer in the mid-term, after setting renewalQuantity to a lower value

Example:

Before AD	AD-7	At AD
<ul style="list-style-type: none"> <li>• Quantity of Acrobat Pro: 30</li> <li>• Renewal quantity: 25</li> </ul>	Mid-term purchase: MOQ 100 <ul style="list-style-type: none"> <li>• Number of licenses of Acro Pro becomes 130</li> <li>• Renewal quantity: 100 (based on the MOQ purchased)</li> </ul>	100 licenses of Acro Pro with MOQ 100 discount are renewed and invoiced.  The customer will be renewed with 100 MOQ because the renewal quantity earlier opted will be overridden by the MOQ quantity

		because the earlier opted renewal quantity is not sufficient to maintain the MOQ.
--	--	---

Scenario 5: Customer purchases MOQ offer in the mid-term after setting renewalQuantity to a higher value

Example:

Before AD	Before AD (For example, last AD+6 months)	At AD
<ul style="list-style-type: none"><li>Quantity of Acrobat Pro: 30</li><li>Renewal quantity: 105</li></ul>	<ul style="list-style-type: none"><li>Mid-term purchase: MOQ 100</li><li>Number of licenses of Acrobat Pro becomes: 130</li></ul>	<p>105 licenses of Acrobat Pro with MOQ 100 discount are renewed and invoiced.</p> <p>The customer will be renewed with 105 licenses of MOQ100 because the renewal quantity of 105 is sufficient for the MOQ offer.</p>

Scenario 6: Customer purchases MOQ offer in the mid-term and later again purchases higher MOQ before renewal

Customers can purchase MOQ offers mid-term, even if they don’t already hold MOQ. The highest MOQ needs to be retained for renewal setting.

Example:

Before AD	Mid-term purchase 1 (Last AD +3 months)	Mid-term purchase 2 (Last AD + 6 months)	At AD
<ul style="list-style-type: none"><li>Customer does not have Acrobat Pro</li></ul>	<ul style="list-style-type: none"><li>Purchased MOQ 100</li><li>Number of Acro Pro licenses: 100</li><li>Renewal quantity: 100</li></ul>	<ul style="list-style-type: none"><li>Purchased MOQ 250</li><li>Number of licenses: 350</li><li>Renewal quantity: 250</li></ul>	<ul style="list-style-type: none"><li>250 licenses will be renewed for the customer with MOQ 250 offer discount.</li></ul>

Manage High Growth Offers through APIs

As explained in the [High Growth Offer workflows](#) section, a reseller and a customer need to perform various steps to provide MOQ offers to customers. This section identifies the APIs necessary to achieve this objective:

- Preview renewal offers
- Update subscription
- Create subscription
- Get subscription
- Create order

Preview renewal offers

Use the PreviewRenewal API to preview the renewal order for the customer. This is the same POST v3/customers/{customer-id}/orders API with orderType as *PREVIEW\_RENEWAL*.

Assumptions:

- You can run the PreviewRenewal API anytime during the current term to get the recommendations.

- The recommendations in the `PreviewRenewal` will be shown only to the customers who hold Acrobat Pro or Standard subscriptions. Also, the recommendation doesn't consider the minimum commit quantity of 3YC.
- Recommendations shown are the same for 3YC and non-3YC Acrobat customers.

Request header

Parameter	Description
X-Request-Id	A unique identifier for the call. The value should be reset for every single request. If this is not provided, then a request ID will be automatically generated. Using a duplicate request ID may return an error.
X-Correlation-Id	<b>Required.</b> A unique identifier for the call. This is to ensure idempotency. In the case of a timeout, the retry call could include the same value. Upon receiving some response, the value should be reset for the next call.
Accept	<b>Required.</b> Specifies the response type. Must be "application/json" for proper usage.
Content-Type	<b>Required.</b> Specifies the request type. Must be "application/json" for proper usage.
Authorization	<b>Required.</b> Authorization token in the form: <code>Bearer &lt;token&gt;</code>
X-Api-Key	<b>Required.</b> The API Key for your integration.

Request body

```
{ "orderType": "PREVIEW_RENEWAL" }
```

Response body

In addition to the standard response of the Order API, the response of the Preview Renewal provides a list of MOQ offers that the customers can avail, under the `eligibleOffers` section:

**Note:** The following code sample includes only the newly added parameters specific to High Growth Offers. For the complete response set, refer to the [Create Order API](#).

```
{
  "referenceOrderId": "",
  "orderId": "",
  "customerId": "1005388836",
  "currencyCode": "USD",
  "orderType": "PREVIEW_RENEWAL",
  "status": "",
  "lineItems":
  [
    {
      "extLineItemNumber": 1,
      "offerId": "65324918CA02A12",
      "quantity": 10,
      "subscriptionId": "a5ea3c7a764545a711d2a153678f02NA",
      "status": ""
    }
  ],
  "eligibleOffers":
```

```
[
  {
    "offerId": "65324918CA14X12",
    "renewalCode": "MOQ_100",
    "eligibilityCriteria": {
      "minQuantity": 100,
      "additionalCriteria": ["THREE_YEAR_COMMIT"],
      "deploymentId": "1450043516",
    },
  },
  {
    "offerId": "65324918CA14Y12",
    "renewalCode": "MOQ_250",
    "eligibilityCriteria": {
      "minQuantity": 250,
      "additionalCriteria": ["THREE_YEAR_COMMIT"],
      "deploymentId": "1450043516",
    },
  },
  {
    "offerId": "65324918CA14Z12",
    "renewalCode": "MOQ_500",
    "eligibilityCriteria": {
      "minQuantity": 500,
      "additionalCriteria": ["THREE_YEAR_COMMIT"]
    },
  },
],
"creationDate": "2024-04-01T07:26:05Z",
}
```

Success response if the customer has already opted for 100 MOQ offer:

```
{
  "referenceOrderId": "",
  "orderId": "",
  "customerId": "1005388836",
  "currencyCode": "USD",
  "orderType": "PREVIEW_RENEWAL",
  "status": "",
  "lineItems": [
    {
      "extLineItemNumber": 1,
      "offerId": "65324918CA14X12",
      "quantity": 100,
      "subscriptionId": "a5ea3c7a764545a711d2a153678f02NA",
      "status": ""
    }
  ],
}
```

```
"eligibleOffers": [
  {
    "offerId": "65324918CA14X12",
    "renewalCode": "MOQ_100",
    "eligibilityCriteria": {
      "minQuantity": 100,
      "additionalCriteria": [
        "THREE_YEAR_COMMIT"
      ],
      "deploymentId": "1450043516"
    }
  },
  {
    "offerId": "65324918CA14Y12",
    "renewalCode": "MOQ_250",
    "eligibilityCriteria": {
      "minQuantity": 250,
      "additionalCriteria": [
        "THREE_YEAR_COMMIT"
      ],
      "deploymentId": "1450043516"
    }
  },
  {
    "offerId": "65324918CA14Z12",
    "renewalCode": "MOQ_500",
    "eligibilityCriteria": {
      "minQuantity": 500,
      "additionalCriteria": [
        "THREE_YEAR_COMMIT"
      ]
    }
  }
],
"creationDate": "2024-04-01T07:26:05Z"
}
```

Response parameter details specific to eligible MOQ offers

The full set of parameters of Order API are available in the [Order resource fields](#) section. The following table lists the parameters specific to High Growth Offers:

Eligible offer parameters

Property	Type	Description	Range/Limits
offerId	string	The unique ID of the offer	Max: 40 characters
renewalCode	string	Unique identifier of the Minimum Order Quantity Offer. Available values are: <ul style="list-style-type: none"><li>MOQ_100</li><li>MOQ_250</li><li>MOQ_500</li></ul>	Max: 30 characters



eligibilityCriteria	Array	The eligibility criteria for availing the MOQ offer.	
minQuantity	Integer	The minimum quantity for which this offer is applicable, also the minimum quantity that the customer needs to commit for a 3YC term to be eligible for this offer. Supported values are: 100, 250, and 500.	Min: 0 Max: 999999
additionalCriteria	string	The additional criteria list for availing the High Growth Offer. Currently, THREE_YEAR_COMMIT is the only supported value, indicating that 3YC is required to avail the High Growth Offers.	Min: 1 item Max: 499 items
deploymentId	string	Unique ID for the deployment.	Max: 40 characters

#### HTTP status codes

Status code	Description
200	Preview Successful
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid customer ID or Order ID

#### Sample response in case of failure

Upon failure, the appropriate HTTP status code based on the reason/type of failure will be sent.

Failure response:

```
{ "code": "4115", "message": "Api Key is invalid or missing" }
```

#### Update subscription

Use the PATCH /v3/customers/{customer-id}/subscriptions/{sub-id}?reset-renewal-code=false API to update the renewal preferences for the customer's subscription with the MOQ offer details.

You can use the optional query param `reset-renewal-code` in the request to remove the `renewalCode` after it has been opted by the customer. Possible values are:

- true
- false

The default value is `false`. For more information, see [sample request and response with query parameter](#).

#### Request header

Same as the [request header given in the previous endpoint](#).

#### Request body

You need to include the `renewalCode` parameter in the auto renewal configuration, as shown in the following code sample:

```
{
  "autoRenewal": {
    "enabled": true,
```

```
    "renewalQuantity": 100,
    "renewalCode": "MOQ_100"
  }
}
```

The renewalCode parameter is the renewal commitment code used for renewal.

Response

The response shows the renewalCode with the selected MOQ offer.

```
{
  "subscriptionId": "a028303a454a168d6b824b6c0dfcc5NA",
  "offerId": "65324918CA14A12",
  "currentQuantity": 10,
  "usedQuantity": 0,
  "autoRenewal": {
    "enabled": true,
    "renewalQuantity": 100,
    "renewalCode": "MOQ_100"
  },
  "creationDate": "2023-09-22T08:38:27Z",
  "renewalDate": "2024-09-22",
  "status": "1000",
  "deployToId": "",
  "currencyCode": "USD",
  "links": {
    "self": {
      "uri":
"/v3/customers/1005388836/subscriptions/a028303a454a168d6b824b6c0dfcc5NA",
      "method": "GET",
      "headers": []
    }
  }
}
```

HTTP status codes

Status code	Description
200	Preview Successful
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid customer ID or Order ID

Sample failure response

On failure, the appropriate HTTP status code based on the reason/type of failure will be sent. For example, if the API key is invalid, the response has status HTTP 403 and the failure response as:

```
{ "code": "4115", "message": "Api Key is invalid or missing" }
```

Use the `/v3/customers/{customer-id}/subscriptions/{sub-id}?reset-renewal-code=true`

**Request body:**

```
{
  "autoRenewal" :{
    "enabled" : true,
    "renewalQuantity" : 25
  }
}
```

**Response:**

```
{
  "subscriptionId": "a028303a454a168d6b824b6c0dfcc5NA",
  "offerId": "65324918CA02A12",
  "currentQuantity": 10,
  "usedQuantity": 0,
  "autoRenewal": {
    "enabled": true,
    "renewalQuantity": 25
  },
  "creationDate": "2023-09-22T08:38:27Z",
  "renewalDate": "2024-09-22",
  "status": "1000",
  "deployToId": "",
  "currencyCode": "USD",
  "links": {
    "self": {
      "uri":
"/v3/customers/1005388836/subscriptions/a028303a454a168d6b824b6c0dfcc5NA",
      "method": "GET",
      "headers": []
    }
  }
}
```

*Create subscription*

Use the POST `v3/customers/{customer-id}/subscriptions` API to create a scheduled subscription for the specific customer.

For example, if the customer has only Acrobat Standard product and not Acrobat Pro, they can use Create Subscription API to create a scheduled subscription for the MOQ.

**Assumptions:**

- The Create Subscription endpoint now also supports `renewalCode` to be given by a partner in a request for opting for a MOQ.

- If the customer accepts to renew with Acrobat Pro MoQ SKU while the customer currently has Acrobat Standard SKU, the Distributor executes create subscription API that allows a scheduled subscription to be created to be activated at the anniversary date.
- For global customers, the following parameters are required:
  - `currencyCode`
  - `deploymentId`

These are part of the Get Subscription API as well.

Request header

Same as the [request header given in the previous endpoint](#).

Request body

```
{
  "offerId": "65322450CA14X12",
  "autoRenewal": {
    "renewalQuantity": 100,
    "renewalCode": "MOQ_100"
  }
}
```

Response

```
{
  "subscriptionId": "e0b170437c4e96ac5428364f674dffNA",
  "offerId": "65322450CA14X12",
  "currentQuantity": 0,
  "usedQuantity": 0,
  "autoRenewal": {
    "enabled": true,
    "renewalQuantity": 100,
    "renewalCode": "MOQ_100"
  },
  "creationDate": "2023-07-18T05:20:19Z",
  "renewalDate": "2024-07-18",
  "status": "1009", // Scheduled
  "currencyCode": "USD",
  "links": {
    "self": {
      "uri":
"/v3/customers/1005331016/subscriptions/e0b170437c4e96ac5428364f674dffNA",
      "method": "GET",
      "headers": []
    }
  }
}
```

HTTP status codes

Status code	Description
200	Created subscription successfully

400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid customer ID or Order ID

### Get subscription

Use the GET `/v3/customers/{customerId}/subscriptions` API to get details of all subscriptions of a customer. Along with other details, it displays the `renewalCode` in the `autoRenewal` preferences if the customer will receive the MOQ offer in the next term.

**Note:** You can use the GET `/v3/customers/<customer-id>/subscriptions/<subscription-id>` API to get MOQ offer details of a specific subscription.

### Request header

Same as the [request header given in the previous endpoint](#).

### Request body

None.

### Response

```
{
  "totalCount": 1,
  "items": [
    {
      "subscriptionId": "43b889db7b4e7aa2d42b54b9813eebNA",
      "offerId": "65322651CA14X12",
      "currentQuantity": 10,
      "usedQuantity": 4,
      "autoRenewal": {
        "enabled": true,
        "renewalQuantity": 10,
        "renewalCode": "MOQ_100"
      },
      "creationDate": "2024-10-23T10:23:11Z",
      "renewalDate": "2025-10-23",
      "status": "1000",
      "currencyCode": "USD",
      "links": {
        "self": {
          "uri":
            "/v3/customers/D1005038400/subscriptions/43b889db7b4e7aa2d42b54b9813eebNA",
          "method": "GET",
          "headers": []
        }
      }
    }
  ],
  "links": {
```

```

    "self": {
      "uri": "/v3/customers/D1005038400/subscriptions",
      "method": "GET",
      "headers": []
    }
  }
}

```

#### HTTP status codes

Status code	Description
200	Successful
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid customer ID or Order ID

#### Create order

No change in the API request or response. Read more about [how to create order through API](#). Customers can send an `offerId` corresponding to MOQ, for example, 65322450CA14X12, to create a mid-term purchase order of MOQ.

### Migrate High Volume Discount customers from VIP to VIP Marketplace

Three-year commitment (3YC) customers benefiting High Volume Discounts in the VIP buying program can migrate to the VIP Marketplace. Only the Acrobat customers in COM (commerce) and GOV (government) market segments and on discount level 17-22 can be migrated to VIP Marketplace. These customers will be reassigned to discount levels 13 or 14 in the VIP Marketplace to ensure consistency in discount offerings.

The discount rates after migration depend on their current Acrobat subscription and pricing plan, as shown below:

#### Customers on pre-pricing action pricing (3YC prior to Sep 1<sup>st</sup>, 2023)

Product	Current 3YC	→	1 <sup>st</sup> 3YC Recommit	→	2 <sup>nd</sup> 3YC Recommit
Acrobat Pro	Honor current pre-pricing action L17-22 price and migrate the users to discount level 13 or 14.		<b>Step down 1:</b> Provide post-pricing action L17-22 price		<b>Step down 2:</b> Recommend and provide eligible High Growth Offers
Acrobat Std	Honor current L17-22 price		Provide post-pricing action L17-22 price		Regular Pricing

#### Customers on post-pricing action pricing (3YC on or after Sep 1<sup>st</sup>, 2023)

Product	Current 3YC	→	1 <sup>st</sup> 3YC Recommit	→	2 <sup>nd</sup> 3YC Recommit
Acrobat Pro	Honor current post-pricing action L17-22 price and migrate the users to discount level 13 or 14.		<b>Step down 1:</b> Recommend and provide eligible High Growth Offers		Regular Pricing
Acrobat Std	Honor current L17-22 price		Regular Pricing		Regular Pricing

Customers with 50-99 licenses are migrated to discount level 13 and with 100 + licenses are migrated to discount level 14 in VIP MP.

Migration workflow

Prerequisites

To migrate high-volume discounts from VIP to VIP MP, the following eligibility criteria must be met:

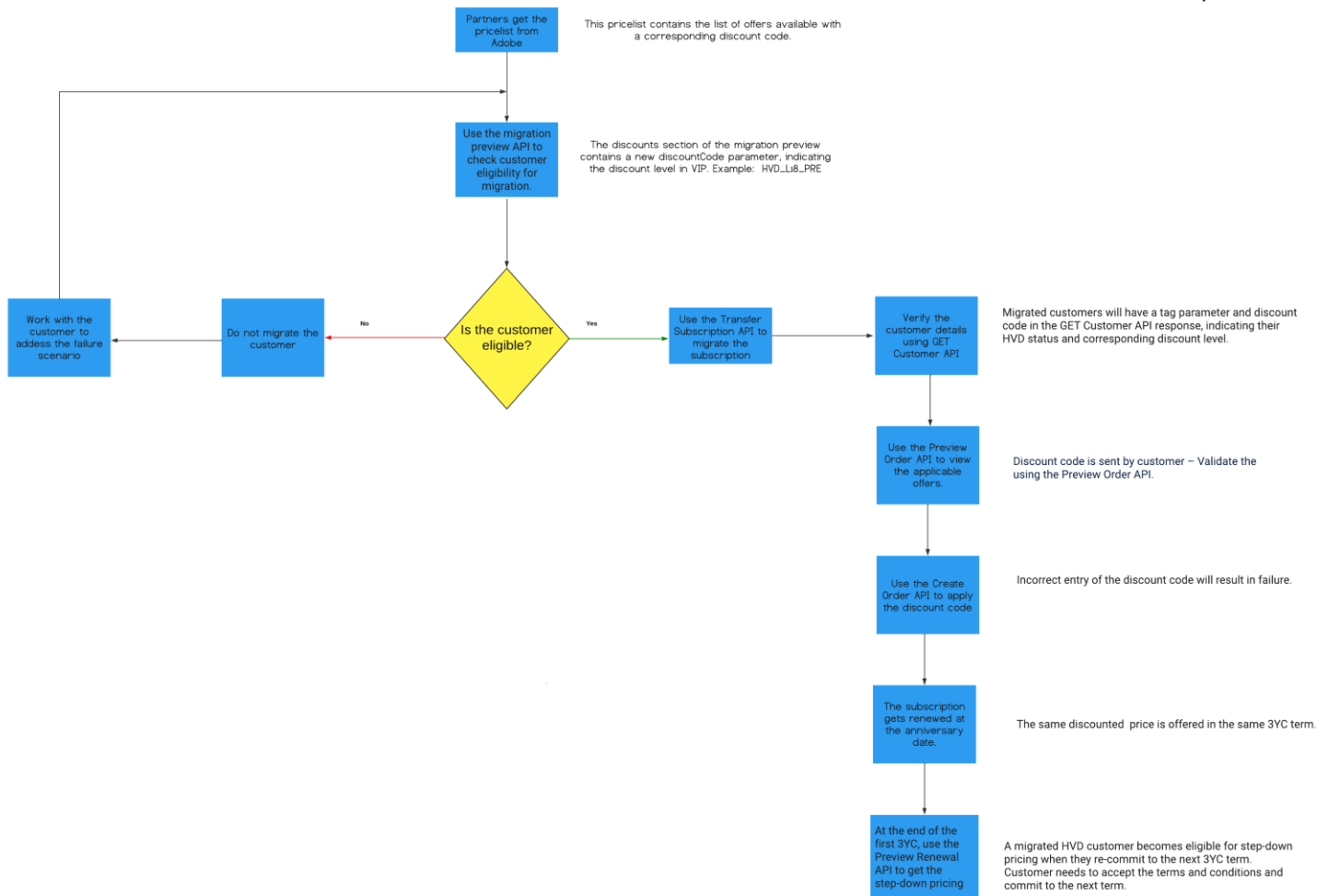
- The customer must be enrolled in a three-year commitment (3YC).
- The number of licenses must be > Minimum Committed Quantity (MCQ) of 3YC. If required, the MCQ must be updated in VIP prior to migration.
- Adobe provides a price list with partners. Each HVD Acrobat Pro subscription in the price list includes a discount code, indicating the corresponding discount level for VIP customers. For instance, if the HVD customer migrating to VIP MP is at discount level 18, the partner must offer a corresponding price that ensures the same discount post-migration. The following discount codes are available:

Discount codes for customers on pre-price actions pricing	Discount codes for customers on post-price actions pricing
HVD_L17_PRE	HVD_L17_POST
HVD_L18_PRE	HVD_L18_POST
HVD_L19_PRE	HVD_L19_POST
HVD_L20_PRE	HVD_L20_POST
HVD_L21_PRE	HVD_L21_POST
HVD_L22_PRE	HVD_L22_POST

**Note:** If you are on pre-price action pricing and committing to the next 3YC term, you need to use the corresponding post-price action pricing discount code for the next 3YC term.

Migration steps

The workflow for migrating an HVD customer from VIP to VIP Marketplace is illustrated in the following figure:



## HVD customer migration using API

### 1. Preview migration

Use the Migration Preview (`/v3/memberships/{membershipId}/offers`) API to verify whether the customer is eligible for migration. This API returns a discount code, indicating the customer's eligibility and applicable discount level.

Sample response:

```

{
  ...
  "benefits": [
    ...
  ],
  "discounts": [
    {
      "level": "14",
      "offerType": "LICENSE",
      "discountCode": "HVD_L18_PRE"
    }
  ]
}

```

**Note:** Refer to the [Migration Preview](#) section for the full set of response.

The newly introduced discount code parameter indicates the discount level and whether the existing subscription is based on the pre-price action pricing or post-price action pricing. For example, HVD\_L18\_PRE



specifies that the HVD customer is at discount level 18 and received the pre-price action pricing for the current subscription.

The `discountCode` parameter will have a null value if the customer is ineligible for the migration.

## 2. Transfer subscription

Use the Transfer Subscription API (POST `/v3/memberships/<membership-id>/transfers`) to migrate the subscription from VIP to VIP MP. Read more about [Transfer Subscription API](#).

## 3. Post migration steps

### a. Get customer details

Partners can use the GET Customer Details API to identify migrated HVD customers distinctly to offer them special pricing for their current and upcoming 3YC terms. The migrated customer consists of newly introduced `tags` and `discountCode` parameters indicating the customer type and discount level.

```
{
  ...
  "tags": ["HVD_MIGRATED_CUSTOMER"]
  "discounts": [
    {
      "level": "12",
      "offerType": "LICENSE",
      "discountCode": "HVD_L18_PRE"
    }
  ]
}
```

**Note:** Refer to the [Get Customer Details](#) section for the full set of response.

### b. Preview Order

Use the Preview Offer API (POST `/v3/orders`) to get the offers available to the migrated customer.

Sample request:

```
{
  "orderType" : "PREVIEW",
  "externalReferenceId" : "759",
  "currencyCode" : "USD",
  "lineItems" : [
    {
      "extLineItemNumber" : 4,
      "offerId" : "80004567EA01A12",
      "quantity" : 1,
      "currencyCode" : "USD",
      "deploymentId" : "12345",
      "discountCode": "HVD_L18_PRE",
    }
  ]
}
```

The `discountCode` can be null, correct, or wrong. Irrespective of your entry, the Preview Offer API returns the correct offer ID as shown in the following sample response:

```
{
  "referenceOrderId" : "",
  "orderType" : "PREVIEW",
}
```

```
{
  "externalReferenceId" : "759",
  "orderId" : "",
  "customerId" : "9876543210",
  "currencyCode" : "USD",
  "creationDate" : "2019-05-02T22:49:54Z",
  "status" : ""
  "lineItems" : [
    {
      "extLineItemNumber" : 4,
      "offerId" : "80004567EA01A12",
      "quantity" : 1,
      "subscriptionId" : "",
      "status" : "",
      "currencyCode" : "USD",
      "deploymentId" : "12345"
      "discountCode": "HVD_L18_PRE",
    }
  ]
}
```

**Note:** Refer to the [Preview Order](#) section for the full set of response.

c. Create Order

Use the Create Order API (POST /v3/orders) to create an order with the discounted price for the migrated customer based on the current discount level.

Sample request:

```
{
  "orderType" : "NEW",
  "externalReferenceId" : "759",
  "currencyCode" : "USD",
  "lineItems" : [
    {
      "extLineItemNumber" : 4,
      "offerId" : "80004567EA01A12",
      "quantity" : 1,
      "currencyCode" : "USD",
      "deploymentId" : "12345"
      "discountCode": "HVD_L18_PRE",
    }
  ]
}
```

The discountCode may be null, correct, or wrong offer ID. The new order will be created only if you provide a valid discount code.

Sample response:

```
{
  "referenceOrderId" : "",
  "orderType" : "NEW",
  "externalReferenceId" : "759",
  "customerId" : "9876543210",
  "orderId" : "5120008001",
  "currencyCode" : "USD",
}
```

```
"creationDate" : "2019-05-02T22:49:54Z",
"status" : "1002",
"lineItems" : [
  {
    "extLineItemNumber" : 4,
    "offerId" : "80004567EA01A12",
    "quantity" : 1,
    "status" : "1002",
    "subscriptionId" : "",
    "currencyCode" : "USD",
    "deploymentId" : "12345"
    "discountCode" : "HVD_L17_PRE",
  }
],
"links" : {...}
}
```

Refer to the [Create Order](#) section for the full set of response.

d. Preview Renewal

Use the Preview Renewal API to get the step-down pricing for the next term.

Sample request:

```
{
  "referenceOrderId": "",
  "orderId": "",
  "customerId": "1005388836",
  "currencyCode": "USD",
  "orderType": "PREVIEW_RENEWAL",
  "status": "",
  "lineItems": [
    {
      "extLineItemNumber": 1,
      "offerId": "65324918CA02A12",
      "quantity": 10,
      "discountCode": "HVD_L17_PRE",
      "subscriptionId": "a5ea3c7a764545a711d2a153678f02NA",
      "status": ""
    }
  ]
}
```

Sample response:

```
{
  "referenceOrderId": "",
  "orderId": "",
  "customerId": "1005388836",
  "currencyCode": "USD",
  "orderType": "PREVIEW_RENEWAL",
```

```
"status": "",
"lineItems":
[
  {
    "extLineItemNumber": 1,
    "offerId": "65324918CA02A12",
    "quantity": 10,
    "discountCode": "HVD_L17_POST",
    "subscriptionId": "a5ea3c7a764545a711d2a153678f02NA",
    "status": "",
  },
],
"creationDate": "2024-04-01T07:26:05Z",
}
```

For example, if a customer on level 17 and on pre-pricing action pricing, this API provides HVD\_L17\_POST as the discount code for the second 3YC term. Read more about [Preview Renewal API](#).

## 6. Create Customer Account

**POST** /v3/customers

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** Customer resource without read-only fields

```
{
  "resellerId" : "5556667778",
  "externalReferenceId" : "342",
  "companyProfile" : {
    "companyName" : "Fairmont",
    "preferredLanguage" : "en-US",
    "marketSegment" : "EDU",
    "address" : {
      "country" : "US",
      "region" : "CA",
      "city" : "San Jose",
      "addressLine1" : "200 Fairmont Ave",
      "addressLine2" : "Apt 123",
      "postalCode" : "95110-1234",
      "phoneNumber" : "800-123-4567"
    },
    "contacts" : [{
      "firstName" : "Donald",
      "lastName" : "Duck",
      "email" : "donald@duck.com",
      "phoneNumber" : "408-123-4567"
    }]
  }
}
```

RESPONSE BODY: Customer resource

```
{
  "externalReferenceId" : "342",
  "customerId" : "9876543210",
  "resellerId" : "5556667778",
  "globalSalesEnabled" : false,
  "companyProfile" : {
    "companyName" : "Fairmont",
    "preferredLanguage" : "en-US",
    "marketSegment" : "EDU",
    "address" : {
      "country" : "US",
      "region" : "CA",
      "city" : "San Jose",
      "addressLine1" : "200 Fairmont Ave",
      "addressLine2" : "Apt 123",
      "postalCode" : "95110-1234",
      "phoneNumber" : "800-123-4567"
    },
    "contacts" : [
      {
        "firstName" : "Donald",
        "lastName" : "Duck",
        "email" : "donald@duck.com",
        "phoneNumber" : "408-123-4567"
      }
    ]
  },
  "discounts" : [{
    "offerType" : "LICENSE",
    "level" : "02",
  }],
  "cotermDate" : "",
  "creationDate" : "2019-05-02T22:49:52Z",
  "status" : "1002",
  "links" : {
    "self" : {
      "uri" : "/v3/customers/9876543210",
      "method" : "GET",
      "headers": []
    }
  }
}
```

HTTP STATUS CODES:

201	Customer account created
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Reseller ID

Assumptions:

- Customer Account id is created by this service and returned synchronously
- Customer payment instrument is managed on the partner marketplace
- cotermdate is normally calculated when a customer's first order is placed
  - Most subscriptions will end or renew on the cotermdate (except for Stock Credit Packs)
- cotermdate can be sent in the request to create an Extended Term customer
- externalReferenceId should be used to pass in the marketplace's Customer ID
  - Optional and does not need to be unique
- Contacts will be admins for the customer's account and receive an admin welcome email
  - Contact names for existing email addresses cannot be changed. If you send a different name for an existing email, that name will be returned in the synchronous response, but it will not persist in our system. The existing first and last name of the contact will be used and returned in any GET calls
- If the Create Customer call succeeds, it will always return a customer with status **1002** (Pending) while final validation happens asynchronously. The account may then become active or inactive. Pending customers can place orders, but they may not update their account using the Update Customer API until the account becomes active or inactive. All orders placed for a pending customer will fail if the customer becomes inactive
- discounts contain a list of objects with the customer's discount levels for each offerType
- Customer can be created with a specified market segment
  - Reseller must be enabled for that market segment
  - If no market segment is sent in the request, the customer will be commercial (COM)

## 7. Get Customer Account

**GET** /v3/customers/<customer-id>

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** None

**RESPONSE BODY:** Customer resource (same as Create Customer)

```
{
  "externalReferenceId": "342""customerId": "9876543210",
  "resellerId": "5556667778",
  "globalSalesEnabled": false,
  "tags": [ "HVD_MIGRATED_CUSTOMER" ]
  "companyProfile": {
    "companyName": "Fairmont",
    "preferredLanguage": "en-US",
    "marketSegment": "EDU",
    "marketSubSegments": [ "K_12"
  ],
  "address": {
    "country": "US",
    "region": "CA",
    "city": "San Jose",
    "addressLine1": "200 Fairmont Ave",
    "addressLine2": "Apt 123",
    "postalCode": "95110-1234",
    "phoneNumber": "800-123-4567"
  },
  "contacts": [
    {
      "firstName": "Mickey",
      "lastName": "Mouse",
      "email": "mickey@mouse.com",
      "phoneNumber": "408-123-4567"
```



```
    }
  ]
},
"discounts": [
  {
    "offerType": "LICENSE",
    "level": "02",
    "discountCode": "HVD_L18_PRE"
  }
],
"cotermDate": "2020-05-02",
"creationDate": "2019-05-02T22:49:52Z",
"status": "1000",
"linkedMembership": {
  "id": "51001315",
  "name": "This is the Group Created for 1005513636",
  "type": "STANDARD",
  "linkedMembershipType": "OWNER",
  "creationDate": "2024-07-17T03:47:35"
},
"links": {
  "self": {
    "uri": "/v3/customers/9876543210",
    "method": "GET",
    "headers": []
  }
}
}
```

**HTTP STATUS CODES:**

200	Customer account details successfully returned
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid customer ID



## 8. Update Customer Account

**PATCH** /v3/customers/<customer-id>

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** CompanyProfile object with optional externalReferenceId

```
{
  "externalReferenceId": "99999",
  "globalSalesEnabled": true,
  "linkedMembership": {
    "type": "STANDARD",
    "name": "This is the Group Created for 1005516130"
  },
  "companyProfile": {
    "companyName": "Fairmont",
    "preferredLanguage": "en-US",
    "marketSegment": "COM",
    "address": {
      "country": "US",
      "region": "CA",
      "city": "San Jose",
      "addressLine1": "200 Fairmont Ave",
      "addressLine2": "Apt 123",
      "postalCode": "95110-1234",
      "phoneNumber": "800-123-4567"
    },
  },
  "contacts": [
    {
      "firstName": "Mickey",
      "lastName": "Mouse",
      "email": "mickey@mouse.com",
      "phoneNumber": "408-123-4567"
    }
  ]
}
```

```
}
```

**RESPONSE BODY:** Same as Get Customer Account Details (Customer resource)

```
{ Customer resource }
```

**Note:** Any contacts specified in this call will receive the admin welcome email. This can be used to resend the email if an end-user did not receive it.

**HTTP STATUS CODES:**

200	Account successfully updated
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer ID

**Assumptions:**

- Update replaces existing data.
- Non-editable fields such as companyName, country, and region need to be passed in but need to match the original values. Otherwise, an error is returned.
- New contacts will be admins for customer's account.
- Any contacts that are removed will still remain admins. To remove admins, a customer admin must use the Adobe Admin Console.
- The name of an existing contact cannot be changed through this API. Customer admin must use the Admin Console to update users' names.
- Customer market segment can be changed through the Update Customer API **only if the customer has no active subscriptions**
  - Reseller must be enabled for the new market segment
- Customer externalReferenceId may now be changed through the Update Customer API.

Benefits with either *commitmentRequest* **OR** *recommitmentRequest* may be sent in the request to:

- Request new 3YC for existing customer
- Request 3YC quantity increase for customer with existing commitment
- Request 3YC *recommitment* for customer with existing commitment
  - Only allowed if the customer has a COMMITTED commitment

9. Get Licenses Pending Partner Order details for Customer

**GET** /v3/customers/<customer-id>/open-acquisitions

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** None

**RESPONSE BODY:**

```
{
  "customerId": "1005375278",
  "resellerId": "1000214906",
  "licenseCount": 2,
  "unpaidLicenses": [
    {
      "referenceId": "PO281097476551717943",
      "creationDate": "2024-03-28T06:03:50Z",
      "expiryDate": "2024-04-04T06:03:50Z",
      "quantity": 1,
      "productName": "Audition"
    },
    {
      "referenceId": "PO281097476551717943",
      "creationDate": "2024-03-28T06:03:51Z",
      "expiryDate": "2024-04-04T06:03:51Z",
      "quantity": 1,
      "productName": "Acrobat Pro"
    }
  ]
}
```

**Note:** This API returns all the licenses for the customer and their corresponding quantities for which partner is yet to place order. Partners can use this API in conjunction with this [notification](#) API to fetch all the products and quantities for each customer for which order has not been placed. The productName returned in the API does not correspond to a specific offer ID. Partners are advised to consult with their customers to identify the appropriate product (including the Team/Enterprise classification) and place orders accordingly.

**HTTP STATUS CODES:**

200	successfully returned open seat addition details
400	Bad request
401	Invalid Authorization token
403	Invalid API Key

## **Deployment Management**

In order for a global customer to purchase offers outside of their home country, they must create and use Deployment resources. If a partner is enabled for global sales, then they are allowed to enable customers for global sales. Once a customer has been enabled for global sales, they can create deployments with addresses outside of the customer's country. These Deployments can then be used to place orders for offers and currencies that are not aligned with the customer's address.

### **10. Create Deployment**

**POST** /v3/customers/<customer-id>/deployments

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** Deployment companyProfile with only address information

```
{
  "companyProfile" : {
    "address" : {
      "country" : "US",
      "region" : "CA",
      "city" : "San Jose",
      "addressLine1" : "200 Fairmont Ave",
      "addressLine2" : "Apt 123",
      "postalCode" : "95110-1234",
      "phoneNumber" : "800-123-4567"
    }
  }
}
```

RESPONSE BODY: Deployment resource

```
{
  "deploymentId": "345434543",
  "companyProfile" : {
    "address" : {
      "country" : "US",
      "region" : "CA",
      "city" : "San Jose",
      "addressLine1" : "200 Fairmont Ave",
      "addressLine2" : "Apt 123",
      "postalCode" : "95110-1234",
      "phoneNumber" : "800-123-4567"
    }
  },
  "creationDate" : "2019-05-02T22:49:52Z",
  "status" : "1000",
  "links" : {
    "self" : {
      "uri" : "/v3/customers/5556667778/deployments/345434543",
      "method" : "GET",
      "headers": []
    }
  }
}
```

HTTP STATUS CODES:

201	Deployment created
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer ID

## 11. Get Deployment

**GET** /v3/customers/<customer-id>/deployments/<deployment-id>

**REQUEST HEADERS:** [See Headers Section](#)

**RESPONSE BODY:** Deployment resource

```
{
  "deploymentId": "345434543",
  "companyProfile" : {
    "address" : {
      "country" : "US",
      "region" : "CA",
      "city" : "San Jose",
      "addressLine1" : "200 Fairmont Ave",
      "addressLine2" : "Apt 123",
      "postalCode" : "95110-1234",
      "phoneNumber" : "800-123-4567"
    }
  },
  "creationDate" : "2019-05-02T22:49:52Z",
  "status" : "1000",
  "links" : {
    "self" : {
      "uri" : "/v3/customers/5556667778/deployments/345434543",
      "method" : "GET",
      "headers": []
    }
  }
}
```



## 12. Update Deployment

**PATCH** /v3/customers/<customer-id>/deployments/<deployment-id>

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** companyProfile or status

```
{
  "status" : "1004",
  "companyProfile" : {
    "address" : {
      "country" : "US",
      "region" : "CA",
      "city" : "San Jose",
      "addressLine1" : "200 Fairmont Ave",
      "addressLine2" : "Apt 123",
      "postalCode" : "95110-1234",
      "phoneNumber" : "800-123-4567"
    }
  }
}
```

**RESPONSE BODY:** Deployment resource

```
{
  "deploymentId": "345434543",
  "companyProfile" : {
    "address" : {
      "country" : "US",
      "region" : "CA",
      "city" : "San Jose",
      "addressLine1" : "200 Fairmont Ave",
      "addressLine2" : "Apt 123",
      "postalCode" : "95110-1234",
      "phoneNumber" : "800-123-4567"
    }
  },
  "creationDate" : "2019-05-02T22:49:52Z",
  "status" : "1000",
  "links" : {
    "self" : {
      "uri" : "/v3/customers/5556667778/deployments/345434543",
      "method" : "GET",
      "headers": []
    }
  }
}
```

**HTTP STATUS CODES:**

201	Deployment created
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer ID

**Assumptions:**

- Address can be updated with this API, but the country cannot be changed
- Status can be updated to de-activate or re-activate a deployment, but a deployment cannot be deactivated if there are any active subscriptions for that deployment

13. Get Deployments for Customer

**GET** /v3/customers/<customer-id>/deployments

**REQUEST HEADERS:** [See Headers Section](#)

**QUERY PARAMETERS:**

param	values	default	description
status	1000   1004	All	
limit	1...100	25	Page size (max number of items to return)
offset	0...N	0	Where to start the page

**RESPONSE BODY:** List of Deployment resources

```
{
  "totalCount" : 9,
  "count" : 3,
  "offset" : 3,
  "limit" : 3,
  "items" : [
    { Deployment resource },
    { Deployment resource },
    { Deployment resource }
  ],
  "links" : {
    "self" : {
      "uri" : "/v3/customers/<customer-id>/deployments?offset=3&limit=3",
      "method" : "GET",
      "headers" : []
    },
    "next" : {
      "uri" : "/v3/customers/<customer-id>/deployments?offset=6&limit=3",
      "method" : "GET",
      "headers" : []
    },
    "prev" : {
      "uri" : "/v3/customers/<customer-id>/deployments?offset=0&limit=3",
      "method" : "GET",
      "headers" : []
    }
  }
}
```

Reseller Change Process

Customers can opt to switch to a different reseller and continue with the new reseller manage their subscriptions and new orders.

Reseller Change is a customer-initiated process where a customer org admin logs-in to Admin Console and checks for the current Reseller Account Manager under the Account Tab.

Admin Console

Overview

Products

Users

Account

Insights

Settings


Account overview

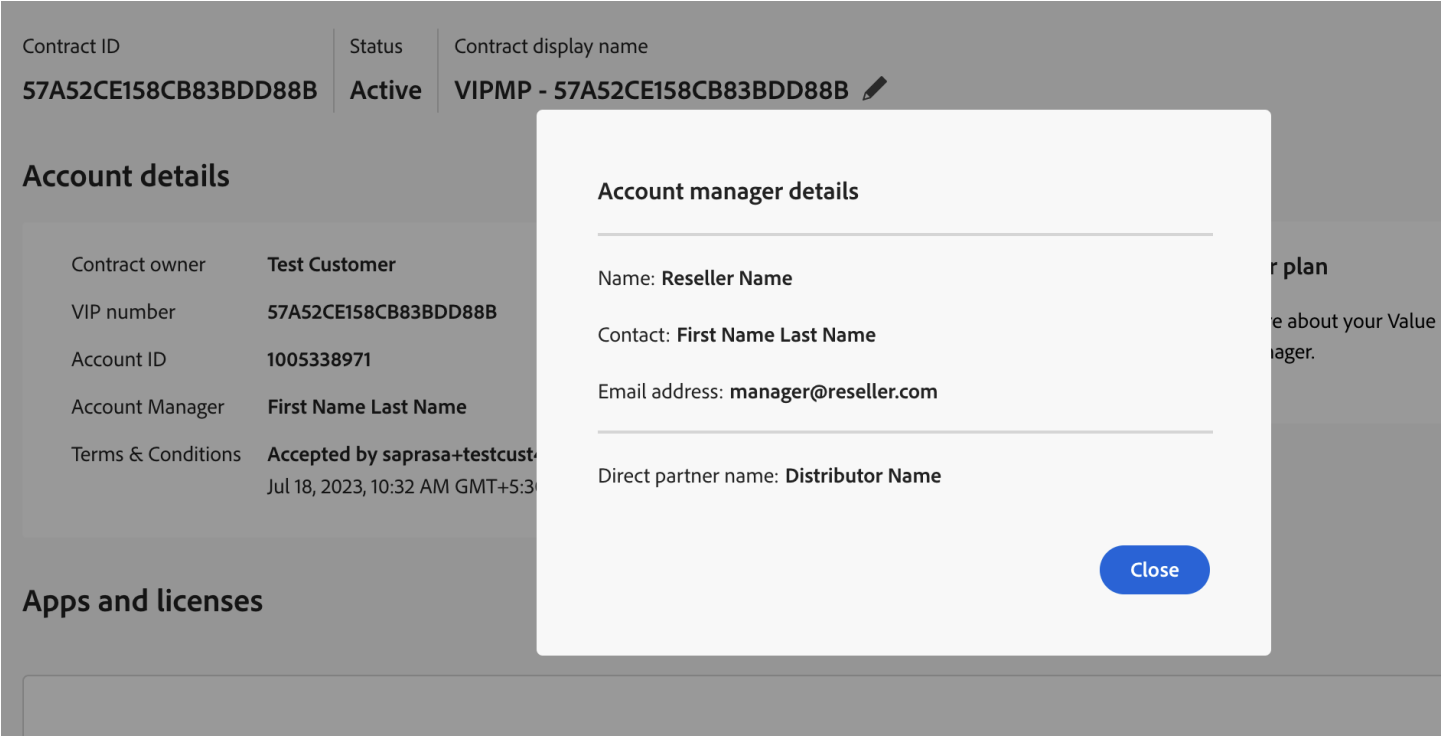
Account overview


Contract ID	Status	Contract display name
57A52CE158CB83BDD88B	Active	VIPMP - 57A52CE158CB83BDD88B

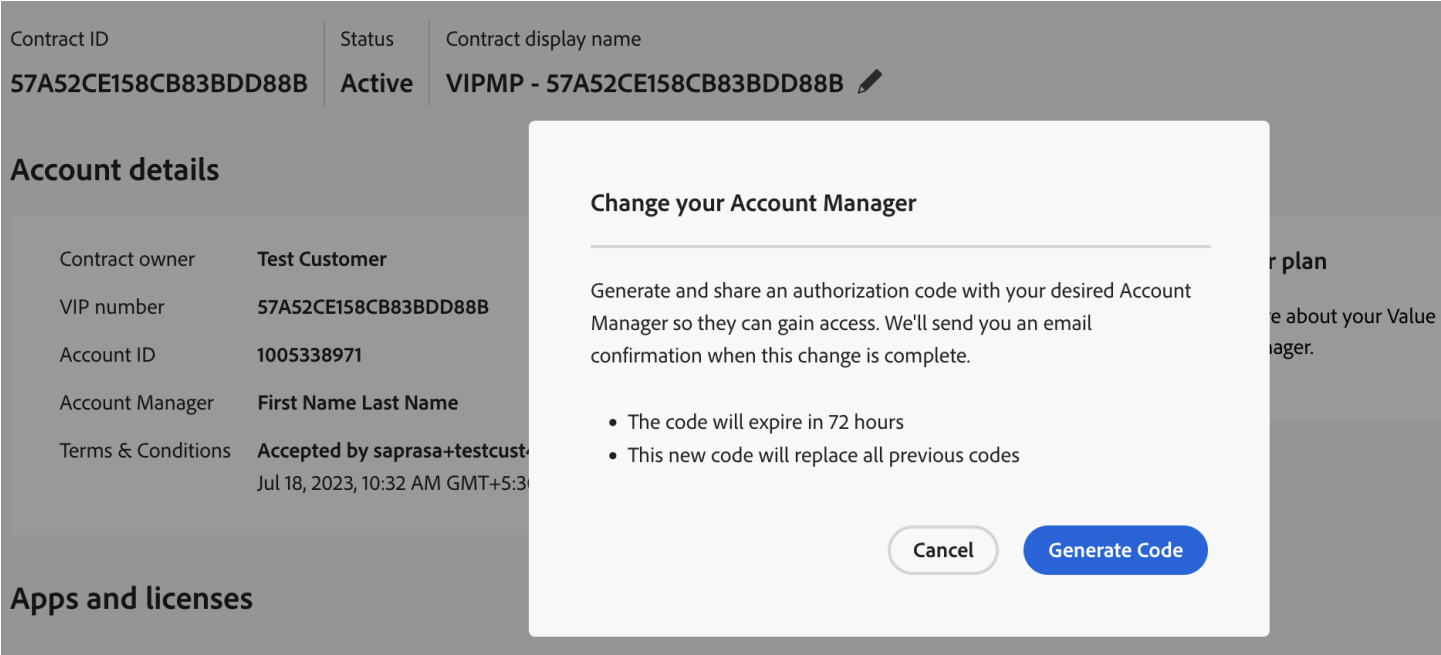
Account details

Contract owner	Test Customer	
VIP number	57A52CE158CB83BDD88B	
Account ID	1005338971	
Account Manager	First Name Last Name	
Terms & Conditions	Accepted by saprasa+testcust4@adobetest.com (Test Customer) Jul 18, 2023, 10:32 AM GMT+5:30	

Account Manager from the Reseller Organization is shown in the account details. To view the Reseller Organization name and the Distributor details, a view button  is provided next to the name. A dialog box opens showing the Reseller and Distributor information when the view button is clicked.



The edit button  next to the view button allows the customer admin to opt for Reseller Change by generating an 8-digit unique code which serves as an approval code for the new reseller to move the customer to his organization.



A new approval code which is valid for 72 hours is generated when **Generate Code** button is clicked.

Contract ID

57A52CE158CB83BDD88B

Status

Active

Contract display name

VIPMP

Account details

Contract owner

Test Customer

VIP number

57A52CE158CB83BDD88B

Account ID

1005338971

Account Manager

First Name Last Name

Terms & Conditions

Accepted by saprasa+testcust  
Jul 18, 2023, 10:32 AM GMT+5:3

Apps and licenses

Change your Account Manager

Generate and share an authorization code with your desired Account Manager so they can gain access. We'll send you an email confirmation when this change is complete.

The code will expire in 72 hours

This new code will replace all previous codes

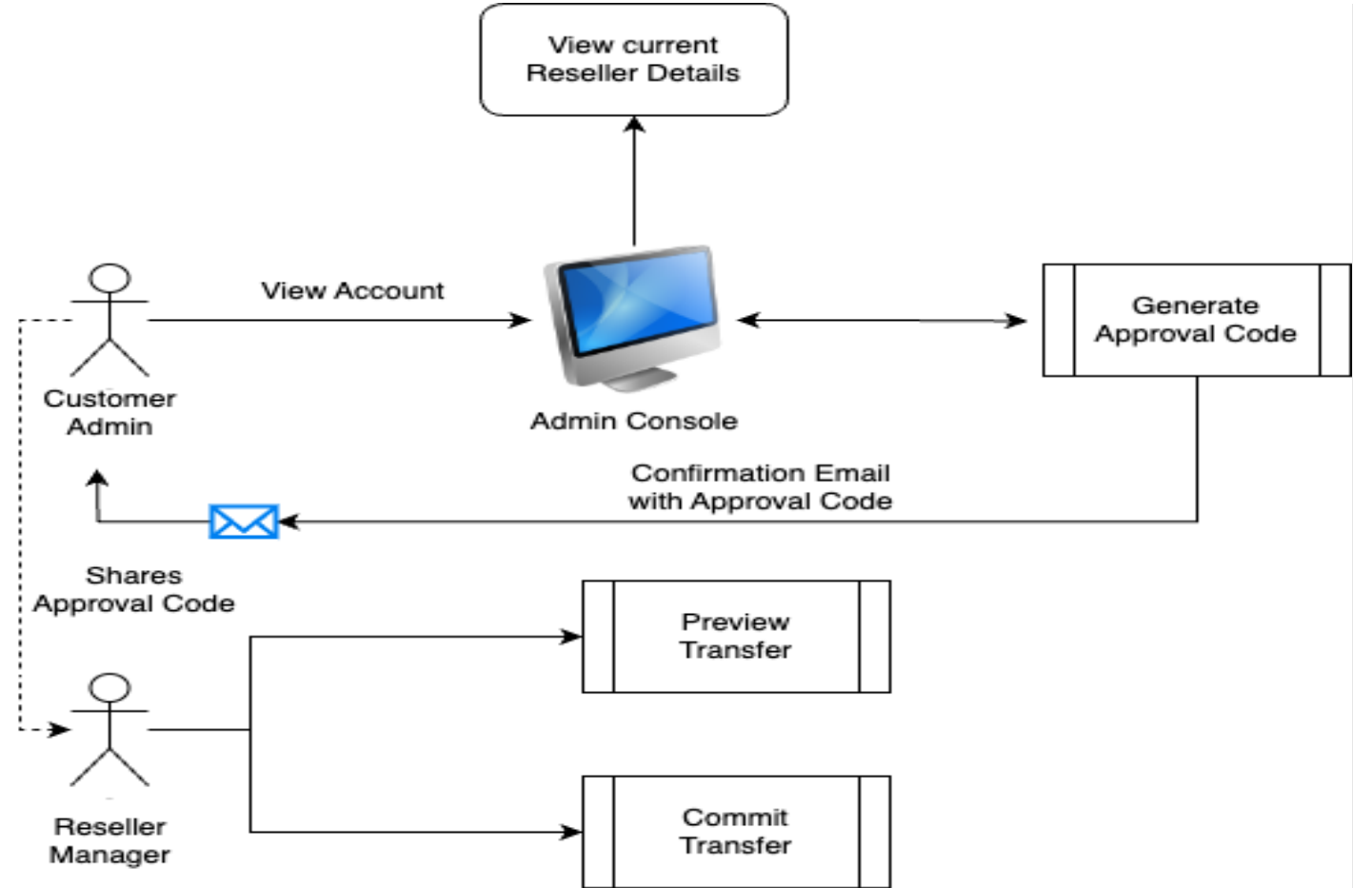
98952071

Copy code

Close

The same code is sent to all the customer admins through email. The customer admin then shares the approval code with the new reseller for the transfer authorization.

If



## 14. Preview Transfer

**POST** /v3/transfers

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:**

```
{
  "type": "RESELLER_CHANGE",
  "action": "PREVIEW",
  "approvalCode" : "8318322",
  "resellerId" : "1000187468",
  "requestedBy" : "customer-admin@email.com"
}
```

**RESPONSE BODY:**

```
{
  "transferId": "",
  "customerId": "1005472660",
  "resellerId": "1000187468",
  "approval": {
    "code": "62159861",
    "expiry": "2024-07-06T06:04:01Z"
  },
  "creationDate": "2024-07-03T06:04:01Z",
  "status": "1002",
  "totalCount": 1,
  "lineItems": [
    {
      "lineItemNumber": 1,
      "offerId": "65304479CA01A12",
      "quantity": 110,
      "subscriptionId": "9bec01597a466898af170a5a203bb1NA",
      "renewalDate": "2025-06-10T16:22:08.000+00:00"
    }
  ],
  "benefits": [
    {
      "type": "THREE_YEAR_COMMIT",
      "commitment": {
        "startDate": "2024-06-10",
        "endDate": "2027-06-09",
        "status": "COMMITTED",
        "minimumQuantities": [
          {
            "offerType": "LICENSE",
            "quantity": 10
          }
        ]
      }
    },
    {
      "commitmentRequest": null,
      "recommitmentRequest": null
    }
  ],
  "discounts": [
    {
      "level": "12",
      "offerType": "LICENSE"
    }
  ]
}
```



15.Commit Transfer

**POST** /v3/transfers

**REQUEST HEADERS:** [See Headers Section](#)

**QUERY PARAMETERS:**

param	values	default	description
accept_customer_with_3yc	true false	true	If customer is 3yc and this flag is true, then reseller change will be executed, otherwise not

**REQUEST BODY:**

```
{
  "type": "RESELLER_CHANGE",
  "action": "COMMIT",
  "approvalCode" : "8318322",
  "resellerId" : "1000187468",
  "requestedBy" : "customer-admin@email.com"
}
```

**RESPONSE BODY:**

```
{
  "transferId": "110014510",
  "customerId": "1005472660",
  "resellerId": "1000187468",
  "creationDate": "2024-07-03T06:04:01Z",
  "status": "1000",
  "totalCount": 1,
  "lineItems": [
    {
      "lineItemNumber": 1,
      "offerId": "65304479CA01A12",
      "quantity": 110,
      "subscriptionId": "9bec01597a466898af170a5a203bb1NA",
      "renewalDate": "2025-06-10T16:22:08.000+00:00"
    }
  ],
  "benefits": [
    {
      "type": "THREE_YEAR_COMMIT",
      "commitment": {
        "startDate": "2024-06-10",
        "endDate": "2027-06-09",
        "status": "COMMITTED",
        "minimumQuantities": [
          {
            "offerType": "LICENSE",
            "quantity": 10
          }
        ]
      }
    },
    {
      "commitmentRequest": null,
      "recommitmentRequest": null
    }
  ],
  "discounts": [
    {
      "level": "12",
      "offerType": "LICENSE"
    }
  ]
}
```

## 16. Get Transfer

**GET** /v3/transfers/<transfer-id>

**REQUEST HEADERS:** [See Headers Section](#)

**RESPONSE BODY:**

```
{
  "transferId": "110014510",
  "customerId": "1005472660",
  "resellerId": "1000187468",
  "creationDate": "2024-07-03T06:04:01Z",
  "status": "1000",
  "totalCount": 0,
  "lineItems": [],
  "benefits": [
    {
      "type": "THREE_YEAR_COMMIT",
      "commitment": {
        "startDate": "2024-06-10",
        "endDate": "2027-06-09",
        "status": "COMMITTED",
        "minimumQuantities": [
          {
            "offerType": "LICENSE",
            "quantity": 10
          }
        ]
      },
      "commitmentRequest": null,
      "recommitmentRequest": null
    }
  ],
  "discounts": [
    {
      "level": "12",
      "offerType": "LICENSE"
    }
  ]
}
```

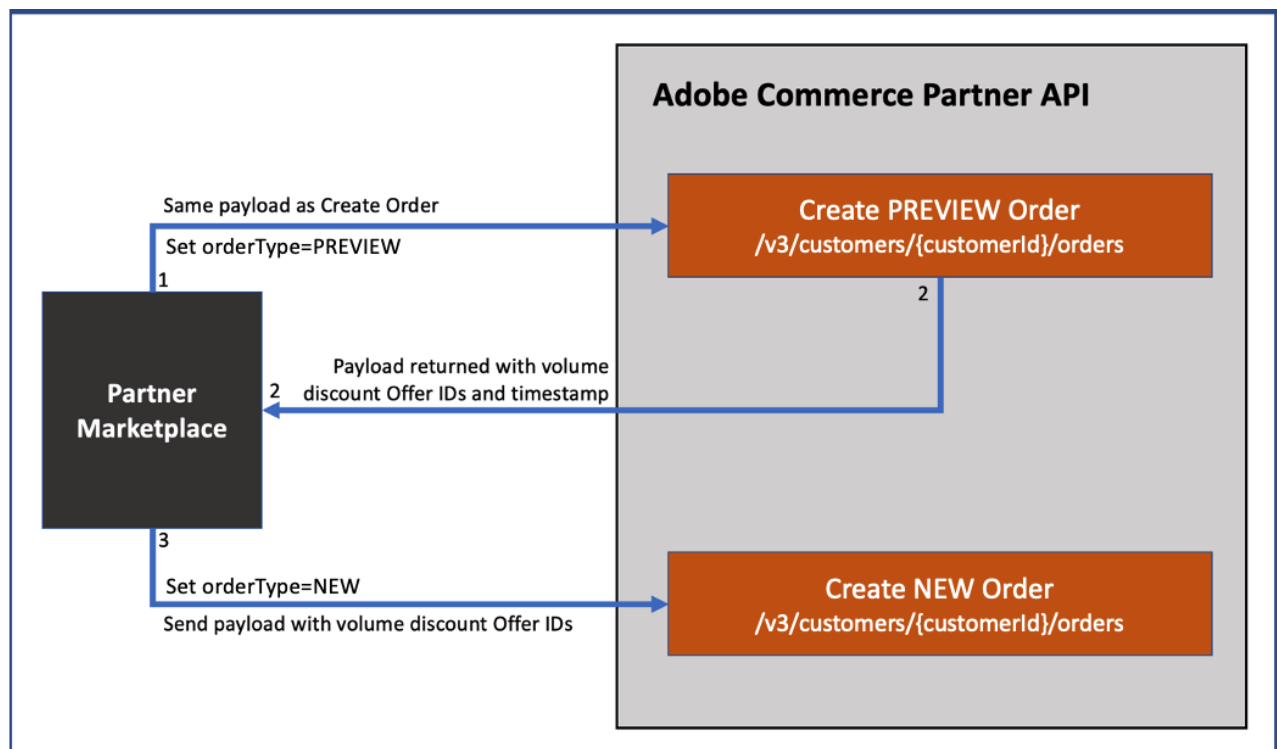
**Note that currently, this API will only return transfers created through the reseller change process (POST /v3/transfers). In subsequent releases, we will allow VIP-VIPMP transfers through these APIs as well (both POST and GET).**

## Order Management

### Volume Discounting

To ensure the best volume discount Offer IDs are used for each order, it is recommended to create an order with type PREVIEW prior to placing an order. Creating a PREVIEW order is an optional call made before placing an order that will determine the best volume discount Offer IDs for the order that the customer is eligible for. The payload is the same as creating a NEW order, and the response returns that payload with the best available offer IDs, which can then be used to place an order through the Create Order API (if the orderType is changed from PREVIEW to NEW)

- Creating a PREVIEW order does not actually place an order. An order with type NEW will need to be created in order to place an order
- A customer's volume discount level will be evaluated after each order. It is based on the customer's current level and the cumulative quantity of volume-discount-eligible products in that order
- Customers transferring their subscriptions from VIP will retain their existing discount level
- A customer's discount level will be reevaluated at renewal time and can go up or down depending on the total renewal quantity across all their subscriptions



## 17. Create Order

**POST** /v3/customers/<customer-id>/orders

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** Order resource without read-only fields

```
{
  "externalReferenceId": "4b6b9a80-cf98-469c-9927-42bd36a4acf",
  "orderType": "RENEWAL",
  "lineItems": [
    {
      "extLineItemNumber": 1,
      "offerId": "65317571CA01A12",
      "quantity": 1,
      "discountCode": "HVD_L18_PRE",
      "subscriptionId": "701db8eac24346b7db262311394975NA",
      "currencyCode": "GBP",
      "deploymentId": "1400002857"
    }
  ]
}
```

### Assumptions:

- orderType is required in the Create Order request
  - must be NEW, RETURN, PREVIEW, PREVIEW\_RENEWAL, or RENEWAL
- subscriptionId is mandatory in lineitems for
  - orderType RENEWAL
  - orderType PREVIEW\_RENEWAL if line items are present
- referenceOrderId is required for RETURN orders and should not be included for other order types
- currencyCode can now be sent at the lineItem level instead of order level
  - Item level currencyCode is used to support global sales (along with deploymentId)
  - The currencyCode should still be sent at the order level if no item level currencyCode is sent
- The discountCode is applicable only to High Volume Discount customers who have migrated from VIP to VIP MP. You can use the discount code only if their discount level in VIP is between 17 and 22.

**RESPONSE BODY:** Order resource

```
{
  "referenceOrderId": "755",
  "externalReferenceId": "4b6b9a80-cf98-469c-9927-42bd36a4acf",
  "orderId": "9203493324",
  "customerId": "1005559275",
  "currencyCode": "",
  "orderType": "RENEWAL",
  "status": "1002",
  "lineItems": [
    {
      "extLineItemNumber": 1,
      "offerId": "65317571CA01A12",
      "quantity": 1,
      "subscriptionId": "701db8eac24346b7db262311394975NA",
      "status": "1002",
      "deploymentId": "1400002857",
      "currencyCode": "GBP"
    }
  ],
  "creationDate": "2024-09-10T07:13:38Z",
  "links": {
    "self": {
      "uri": "/v3/customers/1005559275/orders/9203493324",
      "method": "GET",
      "headers": []
    }
  }
}
```

## Create Order Scenarios

Action	Sample Request/Response	Notes
NEW	<p><b>REQUEST</b></p> <pre>{   "orderType" : "NEW",   "externalReferenceId" : "759",   "currencyCode" : "USD",   "lineItems" : [     {       "extLineItemNumber" : 4,       "offerId" : "80004567EA01A12",       "quantity" : 1,       "currencyCode" : "USD",       "deploymentId" : "12345"     }   ] }</pre> <hr/> <p><b>RESPONSE</b></p> <pre>{   "referenceOrderId" : "",   "orderType" : "NEW",   "externalReferenceId" : "759",   "customerId" : "9876543210",   "orderId" : "5120008001",   "currencyCode" : "USD",   "creationDate" : "2019-05-02T22:49:54Z",   "status" : "1002",   "lineItems" : [     {       "extLineItemNumber" : 4,       "offerId" : "80004567EA01A12",       "quantity" : 1,       "status" : "1002",       "subscriptionId" : "",       "currencyCode" : "USD",       "deploymentId" : "12345"     }   ],   "links" : {...} }</pre>	<ul style="list-style-type: none"> <li>The subscriptionId may be blank while the order is processing. Once the order is complete (status 1000), then all subscriptionIds will be populated and will not change</li> <li>Create order will be used to add seats for the current term</li> <li>Order ID is created by this service and returned synchronously</li> <li>Order may fail asynchronously, and status will be updated to reflect the failure</li> </ul>

<b>RETURN</b> (Cancellation)	<div><b>REQUEST</b><pre>{   "referenceOrderId" : "0123456789",   "orderType" : "RETURN",   "externalReferenceId" : "759",   "currencyCode" : "USD",   "lineItems" : [     {       "extLineItemNumber" : 4,       "offerId" : "80004567EA01A12",       "quantity" : 1,       "currencyCode" : "USD",       "deploymentId" : "12345"     }   ] }</pre></div> <div><b>RESPONSE</b><pre>{   "referenceOrderId" : "0123456789",   "orderType" : "RETURN",   "externalReferenceId" : "759",   "orderId" : "911000833",   "customerId" : "9876543210",   "currencyCode" : "USD",   "creationDate" : "2019-05-02T22:49:54Z",   "status" : "1002",   "lineItems" : [     {       "extLineItemNumber" : 4,       "offerId" : "80004567EA01A12",       "quantity" : 1,       "subscriptionId" : "",       "status" : "1002",       "currencyCode" : "USD",       "deploymentId" : "12345"     }   ],   "links" : {...} }</pre></div>	<ul style="list-style-type: none"><li>referenceOrderId is the order that is being returned</li><li>referenceOrderId must be a valid, <i>returnable</i> order<ul style="list-style-type: none"><li>Either <b>NEW</b> or <b>RENEWAL</b></li></ul></li><li>Line items being returned must match extLineItemNumber, offerId, and quantity of original order<ul style="list-style-type: none"><li>Line items can be cancelled independently in the same or different RETURN order</li><li>No partial line item cancellations</li></ul></li><li>Same 1000   1002   1004 statuses</li><li>As line items from an order get cancelled, the line item status on the original order will change from 1000 to 1008<ul style="list-style-type: none"><li>Only once all line items for an order are cancelled, then the original order will change status to 1008</li></ul></li></ul>
---------------------------------	---	---



**PREVIEW****REQUEST**

```
{
  "orderType" : "PREVIEW",
  "externalReferenceId" : "759",
  "currencyCode" : "USD",
  "lineItems" : [
    {
      "extLineItemNumber" : 4,
      "offerId" : "80004567EA01A12",
      "quantity" : 1,
      "currencyCode" : "USD",
      "deploymentId" : "12345"
      "discountCode": "HVD_L18_PRE",
    }
  ]
}
```

**RESPONSE**

```
{
  "referenceOrderId" : "",
  "orderType" : "PREVIEW",
  "externalReferenceId" : "759",
  "orderId" : "",
  "customerId" : "9876543210",
  "currencyCode" : "USD",
  "creationDate" : "2019-05-02T22:49:54Z",
  "status" : ""
  "lineItems" : [
    {
      "extLineItemNumber" : 4,
      "offerId" : "80004567EA01A12",
      "quantity" : 1,
      "subscriptionId" : "",
      "status" : "",
      "currencyCode" : "USD",
      "deploymentId" : "12345"
      "discountCode": "HVD_L18_PRE",
    }
  ]
}
```

- No orderId, subscriptionId, status, links
- Response can be used to place a new order request if the orderType is changed to **NEW**
- Best available offer ID for this customer and this order is returned
  - Input Offer ID can be any level representing the same product
  - If the Offer IDs in the request are a better discount than customer is allowed, then the correct lower-level Offer ID will be returned.
    - For a **NEW** order, the request would be rejected if the customer is not eligible for an Offer ID
- If NEW order would be rejected, then PREVIEW will be rejected with the same error
- The discountCode is applicable only to High Volume Discount customers who have migrated from VIP to VIP MP. You can use the discount code only if their discount level in VIP is between 17 and 22.

PREVIEW_RENEWAL	<div>REQUEST</div> <pre>{   "orderType" : "PREVIEW_RENEWAL" }</pre> <div>OR</div> <div>REQUEST</div> <pre>{   "orderType" : "PREVIEW_RENEWAL"   "lineItems" : [     {       "extLineItemNumber" : 1,       "offerId" : "80004567EA01A12",       "discountCode": "HVD_L18_PRE",       "subscriptionId" : " e0b170437c4e96ac5428364f674dffNA"     }   ] }</pre> <div>RESPONSE</div> <pre>{   "referenceOrderId": "",   "orderId": "",   "customerId": "1005388836",   "currencyCode": "USD",   "orderType": "PREVIEW_RENEWAL",   "status": "",   "lineItems":     [       {         "extLineItemNumber": 1,         "offerId": "65324918CA02A12",         "quantity": 10,         "discountCode": "HVD_L18_PRE",         "subscriptionId": "a5ea3c7a764545a711d2a153678f02NA",         "status": "",       },     ], }</pre>	<ul style="list-style-type: none"> <li>• No orderId, status, links</li> <li>• In case of <b>no</b> lineitems in the request, the response indicates what would be in the RENEWAL order based on the auto-renewal preferences (autoRenewal.enabled and autoRenewal.renewalQuantity) on the customer's subscriptions</li> <li>• In case of lineitems in the request, the response indicates what would be in the RENEWAL order initiated after anniversary date for the selected line items</li> <li>• If the customer does not have any subscriptions with autoRenewal enabled, then an error will be returned</li> <li>• Best available offer IDs for the renewal order are returned.</li> <li>• The eligibleOffers section lists the High Growth Offers available for the customer. Read more about the <a href="#">High Growth Offers</a> and <a href="#">PreviewRenewal API</a>.</li> <li>• The discountCode indicates the discount code applicable to the HVD customers migrating from VIP to VIP Marketplace. This parameter does not apply to non-HVD customers.</li> </ul>
-----------------	--	---

```

"eligibleOffers":
[
  {
    "offerId": "65324918CA14X12",
    "renewalCode": "MOQ_100",
    "eligibilityCriteria":
    {
      "minQuantity": 100,
      "additionalCriteria":
["THREE_YEAR_COMMIT"],
      "deploymentId": "1450043516",
    },
  },
  {
    "offerId": "65324918CA14Y12",
    "renewalCode": "MOQ_250",
    "eligibilityCriteria":
    {
      "minQuantity": 250,
      "additionalCriteria":
["THREE_YEAR_COMMIT"],
      "deploymentId": "1450043516",
    },
  },
  {
    "offerId": "65324918CA14Z12",
    "renewalCode": "MOQ_500",
    "eligibilityCriteria":
    {
      "minQuantity": 500,
      "additionalCriteria":
["THREE_YEAR_COMMIT"]
    },
  },
],
"creationDate": "2024-04-01T07:26:05Z",
}

```

**RENEWAL****REQUEST**

```

{
  "externalReferenceId": "4b6b9a80-cf98-469c-
9927-42bd36a4acf",
  "orderType": "RENEWAL",
  "lineItems": [
    {
      "extLineItemNumber": 1,
      "offerId": "65317571CA01A12",

```

- RENEWAL order will be used for late renewals (renewals initiated after anniversary date).
- Subscription ID is REQUIRED in the request line item
- The license quantities must be less than or equal to customer's

	<pre>"quantity": 1, "subscriptionId": "701db8eac24346b7db262311394975NA", "currencyCode": "GBP", "deploymentId": "1400002857" } ]</pre>	<p>current subscription current quantities</p> <ul style="list-style-type: none"><li>• Order ID is created by this service and returned synchronously</li><li>• Partner Marketplaces are expected to check the status of the order for success.</li><li>• You can select the expired subscriptions for manual renewal by using the <b>Get All Subscriptions for a Customer</b> API. Subscriptions that can be selected for manual renewal are indicated by the <code>allowedActions</code>: <code>["MANUAL_RENEWAL"]</code> parameter of the <b>Get All Subscriptions of a Customer</b> API response.</li></ul>
	<p><b>RESPONSE</b></p> <pre>{   "referenceOrderId": "",   "externalReferenceId": "4b6b9a80-cf98-469c-9927-42bd36a4acf",   "orderId": "9203493324",   "customerId": "1005559275",   "currencyCode": "",   "orderType": "RENEWAL",   "status": "1002",   "lineItems": [     {       "extLineItemNumber": 1,       "offerId": "65317571CA01A12",       "quantity": 1,       "subscriptionId": "701db8eac24346b7db262311394975NA",       "status": "1002",       "deploymentId": "1400002857",       "currencyCode": "GBP"     }   ],   "creationDate": "2024-09-10T07:13:38Z",   "links": {     "self": {       "uri": "/v3/customers/1005559275/orders/9203493324",       "method": "GET",       "headers": []     }   } }</pre>	

**HTTP STATUS CODES:**

202	Order accepted
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer ID

## Assumptions:

- Renewals will be managed at the subscription level. No new order is expected from the marketplace. Adobe will create a renewal order on a customer's coterminDate if the customer has subscriptions set to autorenew. The renewal Order ID can be retrieved by parsing the response from the Get Order History API (orderType **RENEWAL**)
- Offer IDs are defined and communicated in published price lists
- Each Offer ID is only valid for certain currencies. The currency code send in the request must match the Offer IDs as well as the currency that the distributor is configured for
- externalReferenceId is used to pass in an ID for financial reconciliation
  - The externalReferenceId does not need to be unique and is an optional field

## Restrictions:

- Maximum quantity per line item:
  - 10,000 for Team products
  - 200,000 for Enterprise products
- Maximum line items per order:
  - 499
- Maximum length of externalReferenceId:
  - 35 characters
- Maximum extLineItemNumber:
  - 999999

18. Get Order Details

**GET** /v3/customers/<customer-id>/orders/<order-id>

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** None

**RESPONSE BODY:** Same as Create Order (Order resource)

```
{
  "externalReferenceId" : "759",
  "orderId" : "0123456789",
  "customerId" : "9876543210",
  "orderType" : "NEW",
  "referenceOrderId" : "",
  "referencedOrderId" : "",
  "currencyCode" : "USD",
  "creationDate" : "2019-05-02T22:49:54Z",
  "status" : "1000",
  "lineItems" : [
    {
      "extLineItemNumber" : 4,
      "offerId" : "80004567EA01A12",
      "quantity" : 1,
      "subscriptionId" : "86756309",
      "status" : "1000",
      "currencyCode" : "USD",
      "deploymentId" : "12345"
    }
  ],
  "links" : {
    "self" : {
      "uri" : "/v3/customers/9876543210/orders/0123456789",
      "method" : "GET",
      "headers" : []
    }
  }
}
```

**HTTP STATUS CODES:**

200	Order details successfully retrieved
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer or Order ID

19.Update Order ExternalReferenceId

**PATCH** /v3/customers/<customer-id>/orders/<order-id>

**REQUEST HEADERS:** [See Headers Section](#)

**REQUEST BODY:** Updated externalReferenceId

```
{
  "externalReferenceId" : "34567"
}
```

**RESPONSE BODY:** Order resource with updated externalReferenceId

```
{
  "externalReferenceId" : "34567",
  "orderId" : "0123456789",
  "customerId" : "9876543210",
  "orderType" : "NEW",
  "referenceOrderId" : "",
  "referencedOrderId" : "",
  "currencyCode" : "USD",
  "creationDate" : "2019-05-02T22:49:54Z",
  "status" : "1000",
  "lineItems" : [
    {
      "extLineItemNumber" : 4,
      "offerId" : "80004567EA01A12",
      "quantity" : 1,
      "subscriptionId" : "86756309",
      "status" : "1000",
      "currencyCode" : "USD",
      "deploymentId" : "12345"
    }
  ]
}
```

**HTTP STATUS CODES:**

200	Order successfully updated
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer or Order ID

**Assumptions:**

- Only externalReferenceId may be updated through this API
- The updated externalReferenceId will be reflected in the monthly reconciliation file only if the update is made prior to the reconciliation file being generated

## 20. Get Order History for Customer

**GET** /v3/customers/<customer-id>/orders

**REQUEST HEADERS:** [See Headers Section](#)

### QUERY PARAMETERS:

param	values	default	description
order-type	NEW   TRANSFER   RENEWAL	All	
reseller-id	Valid reseller IDs	All	Regardless of reseller id, only orders for the partner making the request will be returned
status	1000   1002   1004   1026	All	
reference-order-id	Valid order IDs	All	
offer-id	Valid offer IDs	All	
start-date	2019-05-02 2019-05-02T22:49:54Z	Current term start date	Date without timestamp or with timestamp in Zulu time
end-date	2019-05-02 2019-05-02T22:49:54Z	now	Date without timestamp or with timestamp in Zulu time
limit	1...100	25	Page size (max number of orders to return)
offset	0...N	0	Where to start the page

### Assumptions:

- All parameters included in request (besides limit and offset) will be queried together using AND
- Any parameters that are not included will default to the value in the default column
- For **start-date** and **end-date**, a date without timestamp will be evaluated to midnight UTC
  - Dates with timestamp will only be accepted in ISO-8601 format with "Zulu" (UTC) time zone. This is the same format that all datetimes are in CPAPI responses
- The **order-type**, **status**, **offer-id** parameters can be included multiple times to specify multiple values. Including multiple values for a parameter will OR the values in the query
  - Example: To query for orders that are either status 1000 or 10002:  
/v3/customers/<customer-id>?status=1000&status=1002
- Results will be sorted in descending order by creationDate
- Maximum value for offset is the total number of results. Values higher than the total count will return an error
- If a limit greater than the maximum (100) is sent, it will default to the maximum value (100)
- Links to **next** and **prev** will be included in the response if there is a valid next or previous page in the result set



**RESPONSE BODY:** Collection of Order resources

```
{
  "totalCount" : 9,
  "count" : 3,
  "offset" : 3,
  "limit" : 3,
  "items" : [
    { Order resource },
    { Order resource }
  ],
  "links" : {
    "self" : {
      "uri" : "/v3/customers/<customer-id>/orders?offset=3&limit=3",
      "method" : "GET",
      "headers" : []
    },
    "next" : {
      "uri" : "/v3/customers/<customer-id>/orders?offset=6&limit=3",
      "method" : "GET",
      "headers" : []
    },
    "prev" : {
      "uri" : "/v3/customers/<customer-id>/orders?offset=0&limit=3",
      "method" : "GET",
      "headers" : []
    }
  }
}
```

**HTTP STATUS CODES:**

200	Successfully returned order history
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer ID

## Subscription Management

### Auto-Renewal Scenarios

```
"autoRenewal" : {  
    "enabled" : true,  
    "renewalQuantity" : 7  
}
```

Auto-renewal preferences are managed for each subscription independently through the subscription's `autoRenewal` object. These preferences can be updated through the *Update Subscription Auto-Renewal* endpoint. The **enabled** field determines whether or not a subscription will be autorenewed, and the **renewalQuantity** manages the quantity that will be renewed.

The autorenewal preferences are not evaluated until renewal time. You are free to enable/disable autorenewal and change the quantity throughout the length of the subscription. On the customer's `cotermDate`, all of the customer's subscriptions with autorenewal enabled will renew for the specified quantity.

At renewal time, there are two possible scenarios for each subscription:

- Enabled is set to **false** – subscription will become inactive (status 1004)
- Enabled is set to **true** – subscription will be marked for renewal (status will remain 1000) for the amount specified in `renewalQuantity`. During this brief period while waiting for the renewal order to be fulfilled, the `autoRenewal` preferences may not be updated

In order to ensure all subscriptions are renewed (or not) in accordance with the autorenewal preferences, please **make sure the autorenewal preferences are set at least 3 days before the customer's cotermDate (subscription's renewalDate)**. The customer's subscriptions will be renewed at some point soon after midnight UTC, but the exact time may change or vary by region.

If the `renewalQuantity` is higher than the current ordered quantity, then additional licenses will be provisioned after renewal. If the `renewalQuantity` less than the current ordered quantity, then some licenses will be revoked upon renewal.

Adobe will create a renewal order on a customer's `cotermDate` if the customer has subscriptions set to autorenew. The renewal order details and Order ID can be retrieved by parsing the response from the Get Order History API. This ID can then be used in subsequent Get Order calls.

## 21. Get Subscription Details

**GET** /v3/customers/<customer-id>/subscriptions/<subscription-id>

**HEADERS:** [See Headers Section](#)

**REQUEST BODY:** None

**RESPONSE BODY:** Subscription resource

```
{
  "subscriptionId": "43b889db7b4e7aa2d42b54b9813eebNA",
  "offerId": "65322651CA14X12",
  "currentQuantity": 10,
  "usedQuantity": 4,
  "autoRenewal": {
    "enabled": true,
    "renewalQuantity": 10,
    "renewalCode": "MOQ_100"
  },
  "creationDate": "2024-10-23T10:23:11Z",
  "renewalDate": "2025-10-23",
  "status": "1000",
  "currencyCode": "USD",
  "links": {
    "self": {
      "uri":
"/v3/customers/D1005038400/subscriptions/43b889db7b4e7aa2d42b54b9813eebNA",
      "method": "GET",
      "headers": []
    }
  }
}
```

**Note:** The `renewalCode` parameter will be available in the response only if the customer receives a High Growth Offer on the next Anniversary Date.

```
{
  "subscriptionId": "8675309",
  "currentQuantity": 10,
  "usedQuantity": 2,
  "offerId": "65304470CA01012",
  "autoRenewal": {
    "enabled": true,
    "renewalQuantity": 5
  },
  "renewalDate": "2020-05-20",
  "creationDate": "2019-05-20T22:49:55Z",
  "deploymentId": "12345",
  "currencyCode": "USD",
  "status": "1000",
  "links": {
    "self": {
      "uri": "/v3/customers/9876543210/subscriptions/8675309",
      "method": "GET",
      "headers": []
    }
  }
}
```

The **MANUAL\_RENEWAL** value of `allowedActions` field indicates whether a subscription can be selected for manual renewal. Sample response is as follows:

```
{
  "subscriptionId": "1c3a4517c44cfc925704fe942a79abNA",
  "offerId": "65304479CA01A12",
  "currentQuantity": 2,
  "usedQuantity": 0,
  "autoRenewal": {
    "enabled": false,
    "renewalQuantity": 2
  },
  "creationDate": "2024-09-13T09:22:13Z",
  "renewalDate": "2025-09-13",
  "status": "1004",
  "currencyCode": "USD",
  "allowedActions": [
    "MANUAL_RENEWAL"
  ]
  "links": {
    "self": {
```

```
      "uri":
"/v3/customers/1005610729/subscriptions/1c3a4517c44cfc925704fe942a79abNA",
      "method": "GET",
      "headers": []
    }
  }
}
```

HTTP STATUS CODES:

200	Subscription details successfully retrieved
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer or Subscription ID

Assumptions:

- Each subscription corresponds to a product (or group of products)
- Subscriptions are created for line items upon order creation
  - Link to subscription will be added to LineItem
- New orders of a product with an existing subscription will not create a new subscription
  - Quantity will be added to the existing subscription
- currentQuantity is the current license quantity. May differ from ordered quantity while orders are processing
- usedQuantity is the license/consumable usage count.
- offer id is the base discount level offer id.
- renewalDate will be the customer’s coterminDate for most subscriptions
  - Stock Credit Pack subscriptions will have a renewalDate as 1 year from order date
- allowedActions shows the actions allowed on a subscription. Currently, only MANUAL\_RENEWAL is available as the allowed action.

22. Get All Subscriptions for a Customer

GET /v3/customers/<customer-id>/subscriptions

HEADERS: [See Headers Section](#)

REQUEST BODY: None

RESPONSE BODY: Collection of Subscription resources

```
{
  "totalCount": 1,
  "items": [
    {
      "subscriptionId": "43b889db7b4e7aa2d42b54b9813eebNA",
      "offerId": "65322651CA14X12",
      "currentQuantity": 10,
```

```
        "usedQuantity": 4,
        "autoRenewal": {
            "enabled": true,
            "renewalQuantity": 10,
            "renewalCode": "MOQ_100"
        },
        "creationDate": "2024-10-23T10:23:11Z",
        "renewalDate": "2025-10-23",
        "status": "1000",
        "currencyCode": "USD",
        "links": {
            "self": {
                "uri":
"/v3/customers/D1005038400/subscriptions/43b889db7b4e7aa2d42b54b9813eebNA"
            },
            "method": "GET",
            "headers": []
        }
    },
    "links": {
        "self": {
            "uri": "/v3/customers/D1005038400/subscriptions",
            "method": "GET",
            "headers": []
        }
    }
}
```

**Note:** The `renewalCode` parameter will be available in the response only if the customer receives a High Growth Offer on the next Anniversary Date.

HTTP STATUS CODES:

200	Successfully returned all subscriptions
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer ID

Assertions:

- Active subscriptions for VIP customers will be included
- No parameters/filters are implemented at this time

23.Update Subscription Auto-Renewal

**PATCH** /v3/customers/<customer-id>/subscriptions/<subscription-id>

**HEADERS:** [See Headers Section](#)

**REQUEST BODY:** updated autoRenewal dictionary

```
{
  "autoRenewal": {
    "enabled": true,
    "renewalQuantity": 100,
    "renewalCode": "MOQ_100"
  }
}
```

**RESPONSE BODY:** Subscription resource

```
{
  "subscriptionId": "a028303a454a168d6b824b6c0dfcc5NA",
  "offerId": "65324918CA14A12",
  "currentQuantity": 10,
  "usedQuantity": 0,
  "autoRenewal": {
    "enabled": true,
    "renewalQuantity": 100,
    "renewalCode": "MOQ_100"
  },
  "creationDate": "2023-09-22T08:38:27Z",
  "renewalDate": "2024-09-22",
  "status": "1000",
  "deployToId": "",
  "currencyCode": "USD",
  "links": {
    "self": {
      "uri":
"/v3/customers/1005388836/subscriptions/a028303a454a168d6b824b6c0dfcc5NA",
      "method": "GET",
      "headers": []
    }
  }
}
```

HTTP STATUS CODES:

200	AutoRenewal options updated
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer or Subscription ID

Assumptions:

- Enabled flag can only be set to true or false
- Setting enabled to true without the renewalQuantity field will mark all active licenses in the subscription for autorenewal
- Setting enabled to false with a renewalQuantity will disable autorenewal, the renewalQuantity will be ignored
- The autoRenewal preferences can only be updated for an active subscription
- The autoRenewal object is only evaluated at renewal time (the customer's cotermDate)
- If the renewalQuantity is higher than currentQuantity at renewal time, then the additional licenses will be included in renewal. If it is lower, then licenses will be removed at renewal.
- If the renewalQuantity has been explicitly set, the renewalQuantity will not be increased or decreased by additional orders or cancellations (until subscription becomes inactive)
- Three states of autorenewal:
  - Disabled
  - Enabled with renewal quantity
  - Enabled without explicit renewal quantity (all purchased licenses will be renewed. The renewalQuantity will still be returned with the number of purchased licenses)

**Restrictions:**

- Maximum renewalQuantity:
  - 10,000 for Team products
  - 200,000 for Enterprise products (not supported yet)

**24. Create Subscription****POST** /v3/customers/<customer-id>/subscriptions**HEADERS:** [See Headers Section](#)**REQUEST BODY:**

```
{
  "offerId": "65322450CA14X12",
  "autoRenewal": {
    "renewalQuantity": 100,
    "renewalCode": "MOQ_100"
  }
}
```

**RESPONSE BODY:** Subscription resource

```
{
  "subscriptionId": "e0b170437c4e96ac5428364f674dffNA",
  "offerId": "65322450CA14X12",
  "currentQuantity": 0,
  "usedQuantity": 0,
  "autoRenewal": {
    "enabled": true,
    "renewalQuantity": 100,
    "renewalCode": "MOQ_100"
  },
}
```



```
"creationDate": "2023-07-18T05:20:19Z",
"renewalDate": "2024-07-18",
"status": "1009", // Scheduled
"currencyCode": "USD",
"links": {
  "self": {
    "uri":
"/v3/customers/1005331016/subscriptions/e0b170437c4e96ac5428364f674dffNA",
    "method": "GET",
    "headers": []
  }
}
```

HTTP STATUS CODES:

200	Subscription created successfully
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Customer ID

Assumptions:

- Enabled flag can only be set to true
- renewalQuantity field is mandatory for the subscription creation
- Creation of Subscription is allowed only when a customer intends to add a product (that they are not currently subscribed to) during the next auto-renewal.
- Discount level will be calculated and applied at anniversary date with Renewal order. The subscription becomes active with success renewal.
- Customers with no active subscriptions are not allowed to create subscription
- Eligibility checks are evaluated while accepting the create request
- Creating a subscription is allowed only during current date is between 30 days prior to anniversary date to 3 days prior to anniversary date.
- Global customers can create subscriptions. To do so, they need to include the currencyCode in their request. For subscriptions outside the customer’s home country, both currencyCode and deploymentID must be included in the request, as illustrated in the following example:

```
"lineItems": [
  {
    "extLineItemNumber": 1,
    "offerId": "30008891CA01012",
    "currencyCode": "USD",
    "deploymentId": "1400002289"
    "autoRenewal": {
      "enabled": true,
      "renewalQuantity": 30
    }
  }
]
```

```
    }  
  }  
]
```

## **VIP to VIPMP Seamless Move (Transfers):**

Existing customers can be transferred from VIP to VIPMP. Customer consent is implied by sending the customer's VIP Membership ID in the transfer request. The Preview Offers call is used to check a customer's eligibility for transfer and to preview the offers that would be transferred for their existing VIP subscriptions.

### **Seamless Move Eligibility Rules**

#### **Products**

- Customer must have at least one active VIP product
- For a Customer, each existing VIP product must have an equivalent VIPMP product

#### **Three-Year Commit (3YC)**

- Customers with Three-Year Commit are only eligible for transfer if they have accepted 3YC after March 27<sup>th</sup>, 2023 or their 3YC end date is before their next anniversary date

#### **Membership**

- Customer must have a valid existing VIP Membership ID (indirect or direct)
- Customer admin must have accepted VIP Terms & Conditions
- Customer must not be part of Linked membership group

#### **Renewal Window**

- Customers within renewal window are now allowed to transfer to VIPMP
  - If Customer is in the renewal window prior to their anniversary date, they are only allowed to transfer if they have fully renewed

#### **Region**

- Customer must be in the list of [Supported Countries](#)

#### **Reseller**

- Reseller must have accepted the VIPMP Terms & Conditions
- Reseller must be approved to sell in the country of the customer
- Reseller must be enabled to sell into the customer's market segment

#### **Compliance**

- Customer must not have any unpaid VIP Purchase Orders
- Customer must not have any outstanding VIP Purchase Authorizations
  - This can be bypassed by passing an optional query parameter that will expire all outstanding Purchase Authorizations

#### **Returns**

- Customer must not have any orders that can be returned (as per VIP return policy)
  - This can be bypassed by passing an optional query parameter that will disable rollback to VIP

**Note:** Adobe reserves the right to disallow seamless move based solely on Adobe's discretion.

25.Preview Offers

**GET** /v3/memberships/<membership-id>/offers

**HEADERS:** [See Headers Section](#)

**QUERY PARAMETERS:**

param	values	default	description
ignore-order-return	true   false	false	If true, customers with returnable purchases will be eligible for transfer.
expire-open-pas	true   false	false	If true, customers with open Purchase Authorizations will be eligible for transfer.

**REQUEST BODY:** None

**RESPONSE BODY:**

```
{
  "totalCount" : 1,
  "items" : [
    {
      "offerId" : "12345678CA01A12", // Offer ID for marketplace to use
      "currencyCode" : "USD",
      "quantity" : 10
      "renewalDate" : "2020-06-08"
    }
  ],
  "benefits": [
    {
      "type": "THREE_YEAR_COMMIT",
      "commitment": {
        "startDate": "2024-05-14",
        "endDate": "2027-04-11",
        "status": "ACTIVE",
        "minimumQuantities": [
          {
            "offerType": "LICENSE",
            "quantity": 11
          }
        ]
      },
      "commitmentRequest": {
        "startDate": "2027-04-12",
        "endDate": "2030-04-11",
        "status": "ACCEPTED",
        "minimumQuantities": [
          {
            "offerType": "LICENSE",
            "quantity": 11
          }
        ]
      }
    },
    {
      "type": "THREE_YEAR_COMMIT",
      "commitment": {
        "startDate": "2024-05-14",
        "endDate": "2027-04-11",
        "status": "ACTIVE",
        "minimumQuantities": [
          {
            "offerType": "LICENSE",
            "quantity": 11
          }
        ]
      },
      "commitmentRequest": {
        "startDate": "2027-04-12",
        "endDate": "2030-04-11",
        "status": "ACCEPTED",
        "minimumQuantities": [
          {
            "offerType": "LICENSE",
            "quantity": 11
          }
        ]
      }
    }
  ],
  "discounts": [
    {
      "level": "12",
      "offerType": "3YC"
    }
  ]
}
```

```
{
  "totalCount": 1,
  "items": [
    {
```

```
        "offerId": "12345678CA01A12", // Offer ID for marketplace to use
        "currencyCode" : "USD",
        "quantity": 10,
        "renewalDate": "2020-06-08"
    }
],
"benefits": [
    {
        "type": "THREE_YEAR_COMMIT",
        "commitment": {
            "startDate": "2024-05-14",
            "endDate": "2027-04-11",
            "status": "ACTIVE",
            "minimumQuantities": [
                {
                    "offerType": "LICENSE",
                    "quantity": 11
                }
            ]
        },
        "commitmentRequest": {
            "startDate": "2027-04-12",
            "endDate": "2030-04-11",
            "status": "ACCEPTED",
            "minimumQuantities": [
                {
                    "offerType": "LICENSE",
                    "quantity": 11
                }
            ]
        },
    },
]
},
"discounts": [
    {
        "level": "12",
        "offerType": "3YC",
        "discountCode": "HVD_L18_PRE"
    }
],
"renewalStatus": "No",
}
```

HTTP STATUS CODES:

200	Preview returned
400	Bad request
401	Invalid Authorization token

403	Invalid API Key
404	Invalid Membership ID

[Full set of transfer eligibility error codes](#)

**Assumptions:**

- Each item corresponds to one subscription
- The subscription is active until the renewalDate
  - All subscriptions will have auto-renewal enabled by default upon transfer (so they will renew on the renewalDate)
- If the customer is in the renewal window after their anniversary date and *has not renewed some products*, the non-renewed products will be included with renewalDate in the past. These items will be created as inactive subscriptions – the same behavior as if the customer had products in VIPMP that did not renew
- Benefits and discount will be shown if customer is 3yc customer
- The renewalStatus parameter indicates the renewal status of subscriptions in VIP. The possible values are:
  - **Full:** All subscriptions are renewed before the migration.
  - **Partial:** Not all subscriptions of the customer are renewed.
  - **No:** None of the subscriptions are renewed.

**26.Transfer Subscriptions**

**POST** /v3/memberships/<membership-id>/transfers

**HEADERS:** [See Headers Section](#)

**QUERY PARAMETERS:**

param	values	default	description
ignore-order-return	true   false	false	If true, customers with returnable purchases can be transferred.  Note: setting to true will disable rollback to VIP and the “returnable” purchase can no longer be returned.
expire-open-pas	true   false	false	If true, customers with open Purchase Authorizations can be transferred. Any open Pas will be expired during the async portion of the transfer.

**REQUEST BODY:**

{ "resellerId" : "999888777" }
--------------------------------------

**RESPONSE BODY:**

```
{
  "transferId" : "5555luaigdfads555",
  "customerId" : "",
  "membershipId" : "12345678",
  "resellerId" : "999888777",
  "creationDate" : "2019-12-10T22:49:55Z",
  "status" : "1002",
  "lineItems" : [
    {
      "lineItemNumber" : 1,
      "offerId" : "12345678CA01A12",
      "currencyCode" : "USD",
      "quantity" : 10,
      "subscriptionId" : ""
    }
  ],
  "links" : {
    "self" : {
      "uri" : "/v3/memberships/12345678/transfers/5555luaigdfads555",
      "method" : "GET",
      "headers" : []
    }
  }
}
```

**HTTP STATUS CODES:**

202	Transfer request received/initiated
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Membership ID / Invalid Reseller ID

[Full set of transfer eligibility error codes](#)



27. Get Transfer Details

GET /v3/memberships/<membership-id>/transfers/<transfer-id>

HEADERS: [See Headers Section](#)

REQUEST BODY: None

RESPONSE BODY: Transfer resource

```
{
  "transferId" : "5555luaigdfads555",
  "customerId" : "10008675",
  "membershipId" : "12345678",
  "resellerId" : "999888777",
  "creationDate" : "2019-12-10T22:49:55Z",
  "status" : "1000",
  "lineItems" : [
    {
      "lineItemNumber" : 1,
      "offerId" : "12345678CA01A12",
      "quantity" : 10,
      "subscriptionId" : "8675309",
      "currencyCode" : "USD",
      "deploymentId" : "12345"
    }
  ],
  "links" : {
    "self" : {
      "uri" : "/v3/memberships/12345678/transfers/5555luaigdfads555",
      "method" : "GET",
      "headers" : []
    }
  }
}
```

HTTP STATUS CODES:

202	Transfer request received/initiated
400	Bad request
401	Invalid Authorization token
403	Invalid API Key
404	Invalid Membership or Transfer ID

**Notification Management:**

Partners can fetch different types of resources which will be based on notification type sent in the request. The query params of the API can also vary based on notification type. Currently we only support **LICENSES\_PENDING\_PARTNER\_ORDER** notification type and partner needs to pass **reseller-id** query param for this notification type.

Notification Type	Supported query params	Description
LICENSES_PENDING_PARTNER_ORDER	<div>1. reseller-id=100022361</div> <div>2. limit=10</div> <div>3. offset=0</div>	The <b>LICENSES_PENDING_PARTNER_ORDER</b> notification type returns all the customers under a given reseller who have at least one license created within the last 7 days for which order has not been placed yet by partner.

28.GET LICENSES\_PENDING\_PARTNER\_ORDER Notification

**GET** /v3/notifications?notification-type=LICENSES\_PENDING\_PARTNER\_ORDER&reseller-id=<resellerId>&limit=10&offset=0

**HEADERS:** [See Headers Section](#)

**REQUEST BODY:** None

**RESPONSE BODY:** Notification resource

```
{
  "limit": 10,
  "offset": 0,
  "totalCount": 1,
  "count": 1,
  "items": [
    {
      "customerId": "1005379198",
      "resellerId": "1000177552",
      "notificationType": "LICENSES_PENDING_PARTNER_ORDER",
      "links": {
        "self": {
          "uri": "/v3/customers/1005379198/open-acquisitions",
          "method": "GET",
          "headers": []
        }
      }
    }
  ],
  "links": {
    "self": {
      "uri": "/v3/notifications?notification-type=LICENSES_PENDING_PARTNER_ORDER&reseller-id=1000177552&limit=10&offset=0",
      "method": "GET",
      "headers": []
    }
  }
}
```

**HTTP STATUS CODES:**

200	Successfully returned notification resource
400	Bad request
401	Invalid Authorization token
403	Invalid API Key

## Appendix

### 29. Headers

X-Request-Id	A unique identifier for the call. The value should be reset for every single request. If this is not provided, then a request ID will be automatically generated. Using a duplicate request ID may return an error.
X-Correlation-Id	<b>Required.</b> A unique identifier for the call. This is to ensure idempotency. In the case of a timeout, the retry call could include the same value. Upon receiving some response, the value should be reset for the next call.
Accept	<b>Required.</b> Specifies the response type. Must be "application/json" for proper usage.
Content-Type	<b>Required.</b> Specifies the request type. Must be "application/json" for proper usage.
Authorization	<b>Required.</b> Authorization token in the form "Bearer <token>"
X-API-Key	<b>Required.</b> The API Key for your integration

### 30. Idempotency – Correlation ID Header

The **X-Correlation-Id** is expected for all requests to the API. Where **X-Request-Id** should be unique per request, **X-Correlation-Id** should be unique per *intent*. Intent in this case means the data in the request and the attempted action.

- This means that upon receiving some response from Adobe (success or failure), the intent has been satisfied and the Correlation ID should not be used again.
- However, if no response is received from Adobe (in the case of a timeout), the same Correlation ID could be used again for the retry call (if the retry has the same intent).
- If a response was generated by Adobe for a given correlation ID, any subsequent requests to the same API with the same Correlation ID will receive that same response
- If a call fails with a 400 Bad Request error and you try again after fixing the data, the new call must have a new Correlation ID (as that would be a different intent).
- If the Correlation ID header is not included, a 400 error with status code **4119 – Correlation ID is Invalid or Missing** will be returned. See the Status Codes section below for more info.

Duplicate Request IDs:

- If the Correlation ID is duplicate, Request ID is not evaluated – the original response for that intent will be returned
- If the Correlation ID is different and Request ID is a duplicate, then a 400 error with status code **4120 – Duplicate request id, same requestId has already been processed** will be returned

### 31. HTTP Status Codes

**2xx** – Success

**4xx** – Error in client's request

**5xx** – Error in Adobe servers

200	OK	Everything worked as expected
201	Created	Resource has been successfully created
202	Accepted	Request has been accepted and will be acted on later
400	Bad Request	The request was unable to be completed due to some client error
401	Unauthorized	The authorization token is invalid
403	Forbidden	The API key is invalid
404	Not Found	The requested resource does not exist
406	Item contain deployIds	If order request is with deploymentIds at line item but customer is non-global.
429	Too Many Requests	Too many requests have been made in a brief period. Currently there is no limiting being done, so this status code should not appear.
500	Internal Server Error	There was an error on Adobe's end. Please try request again

## 32. Status Codes & Error Handling

### Resource Status Codes

Status Code	Description	Applicable Resources
1000	<b>Resource Status: Green</b> Account - Active Order - Complete/Filled Subscription - Active	Reseller Account, Customer Account, Order, Subscription Transfer, Deployment
1002	<b>Resource Status: Yellow</b> Account - Pending Order - Open Subscription - Pending	Reseller Account, Customer Account, Order, Subscription Transfer
1004	<b>Resource Status: Red</b> Account - Inactive Order - Failed Subscription - Inactive  If there is another status code that describes the status better, that code will be used instead.	Reseller Account, Customer Account, Order, Subscription Transfer, Deployment
1008	<b>Order Status:</b> Cancelled	Order
1009	<b>Subscription Status:</b> Scheduled	Subscription
1010	<b>Account Status:</b> Inactive <b>Reason:</b> Invalid Address	Reseller Account, Customer Account
1012	<b>Account Status:</b> Inactive <b>Reason:</b> Account is blocked	Reseller Account, Customer Account
1014	<b>Account Status:</b> Inactive <b>Reason:</b> Customer already exists with the same Company Name and primary admin	Customer Account
1020	<b>Order Status:</b> Failed <b>Reason:</b> Distributor is inactive	Order

Status Code	Description	Applicable Resources
1022	<b>Order Status:</b> Failed <b>Reason:</b> Reseller is inactive	Order
1024	<b>Order Status:</b> Failed <b>Reason:</b> Customer is inactive	Order
1026	<b>Order Status:</b> Failed <b>Reason:</b> Customer ID is invalid	Order

## Error Handling

Any Commerce Partner API status response with a **4xx** or **5xx** HTTP status code includes an error message consisting of a code field containing the HTTP response code, and a message field describing the error. Below is an example error response:

### RESPONSE BODY:

```
{
  "code" : "1117",
  "message" : "Some Fields are Invalid"
  "additionalDetails" : [
    "companyProfile.companyName",
    "companyProfile.contacts[0].firstName"
  ]
}
```

Some error responses will include `additionalDetails`, an array of strings with more information on the specific error. These strings *may include* the names of fields with errors, line-item numbers, more specific reasoning, or other information.

Error Code	Message	Applicable API calls	HTTP Status Code
1114	Invalid Distributor	Create Reseller Account	400
1115	Invalid Reseller	Create Customer Account, Get Reseller Account, Update Reseller Account, Reseller Change Preview, Reseller Change Commit	404

Error Code	Message	Applicable API calls	HTTP Status Code
1116	Invalid Customer	Get Customer Account, Update Customer Account, Get Order History, Get All Subscriptions	404
1117	Some fields are invalid  For debugging purposes, the names of the fields will be included in the invalidFields array in the response body.	Create Reseller, Update Reseller, Create Customer, Update Customer, Create Order, Update Subscription Auto-Renewal	400
1118	Invalid Address  <b>Note:</b> If address and other fields are invalid, then <b>1117</b> will be used	Create Reseller, Update Reseller, Create Customer, Update Customer	400
1119	Some fields are not editable	Update Customer, Update Reseller, Cancel Order, Update Subscription Auto-Renewal	400
1121	Request contains additional unexpected fields	All	400
1122	Request is missing required fields	All (except for GETs)	400
1123	Duplicate externalReferenceId	Deprecated	400
1124	Internal Server Error. Please try again later	All	500
1125	Account cannot be updated while status is pending or blocked	Update Reseller, Update Customer	400



Error Code	Message	Applicable API calls	HTTP Status Code
1126	Customer cannot be created because reseller account is inactive	Create Customer	400
1127	Account already exists	Create Customer	400
1128	Invalid Reseller Change Code	Reseller Change Preview, Reseller Change Commit	404
1129	Reseller Change Code is expired	Reseller Change Preview, Reseller Change Commit	404
1131	Reseller Change is already complete	Reseller Change Preview, Reseller Change Commit	400
1132	Invalid value for search parameter	Get Order History (V3)	400
1133	Offset out of range	Get Order History (V3)	400
1134	Invalid operation on path [INVALID_RESELLER_ID: Reseller ID in the input is either invalid or same as the current reseller id]	Reseller Change Preview Reseller Change Commit	400
1139	Partner is not enabled for global sales	Update Customer Reseller Change Commit Create Deployment Create Order	400
1140	Customer is not enabled for global sales	Update Customer Reseller Change Commit Create Deployment Create Order	400

Error Code	Message	Applicable API calls	HTTP Status Code
1141	Deployment and customer country cannot be the same	Create Deployment	400
1142	Market segment not allowed for global sales	Update Customer Create Order	400
1144	Linked Account Not Found	Update Customer Get Customer	404
2115	Invalid Customer or Order ID	Get Order Details, Cancel Order	404
2116	Invalid status	Cancel Order	400
2117	Cancellation window has closed	Cancel Order	400
2118	Order has already been cancelled	Cancel Order	400
2119	Too many line items	Preview Order, Create Order	400
2120	Line item quantity out of range	Preview Order, Create Order	400
2121	Duplicate line item numbers	Preview Order, Create Order	400
2122	Line item Offer ID is invalid	Preview Order, Create Order	400
2123	extLineItemNumber out of range	Preview Order, Create Order	400
2124	Distributor not allowed to sell the program	Preview Order, Create Order	400
2125	Distributor not allowed to sell in the currency for the region	Preview Order, Create Order	400

Error Code	Message	Applicable API calls	HTTP Status Code
2126	externalReferenceId exceeds maximum character limit	Create Order, Create Reseller , Create Customer	400
2128	Currency is not valid for Offer ID	Preview Order, Create Order	400
2129	Customer is not eligible to purchase Offer ID at line item: #	Create Order	400
2130	Line item Offer ID does not match original order	Create Return Order (V3)	400
2131	extLineItemNumber does not match original order	Create Return Order (V3)	400
2132	Line item quantity does not match original order	Create Return Order (V3)	400
2133	Line item has already been returned	Create Return Order (V3)	400
2134	Line item cannot be returned as the order already expired	Create Return Order (V3)	400
2135	Invalid market segment used for customer	Create Reseller Update Reseller Create Customer Update Customer	400
2136	Please review the renewal settings. Auto-renewal needs to be turned on for at least 1 quantity	Create PREVIEW_RENEWAL Order	400
2137	Currency code not aligned	Create Order	400
2138	Invalid Deployment	Create Order	404

Error Code	Message	Applicable API calls	HTTP Status Code
2139	Deployment not allowed for this contract	Create Order	400
2140	Order contains order level currency and line-item level currency	Create Order	400
3115	Invalid Customer or Subscription ID	Get Subscription Details, Update Subscription Auto-Renewal	404
3116	renewalQuantity out of range	Update Subscription Auto-Renewal	400
3117	currentQuantity out of range		400
3118	orderQuantity out of range		400
3119	Inactive Subscription is not Editable	Update Subscription Auto-Renewal	400
3120	Update could not be performed because it would create an invalid renewal order	Update Subscription Auto-Renewal	400
3121	Subscription Not allowed for Renewal	Renewal Order after Anniversary Date	400
3122	Not within Renewal Window	Renewal Order attempted 30 days after Anniversary Date / Subscription Creation request not in Renewal Window	400
3123	Product Expired	Line Item offer id has expired	400
3124	Subscription Already Exists	Subscription already exists for the Offer ID	400
3125	No Subscriptions	No subscriptions exist for the customer	400

Error Code	Message	Applicable API calls	HTTP Status Code
3126	Global Customer Unsupported	This feature is not supported for Global Customer	400
4115	API key is invalid or missing	All [Adobe I/O]	403
4116	Authorization token is invalid	All [Adobe I/O]	401
4117	Authorization token is missing	All [Adobe I/O]	403
4118	Too many requests	All [Adobe I/O]	429
4119	Correlation ID is Invalid or Missing	All	400
4120	Duplicate request id, same requestId has already been processed	All	400
4121	You are not allowed to perform that action on this resource	All	403
5115	Invalid Membership ID	Preview Offers, Transfer Subscriptions	400
5116	Invalid Membership or Transfer ID	Get Transfer Details	400
5117	Customer is not eligible for transfer (see table below for REASON_CODE list)	Preview Offers, Transfer Subscriptions	400
5118	Customer has already been transferred	Preview Offers, Transfer Subscriptions	400
5119	Customer cannot be transferred because reseller account is inactive	Transfer Subscriptions	400
5120	Customer cannot be transferred because there are no admin contacts	Preview Offers, Transfer Subscriptions	400

Error Code	Message	Applicable API calls	HTTP Status Code
5121	Transfer currently in progress for this customer	Preview Offers, Transfer Subscriptions	400
5122	Order placement is currently unavailable. Please try again later.	Create Order	400
5123	Invalid Discount Code		400 <b>Description:</b> The discount code is not in use
5124	Invalid Renewal Quantity for Discount Code		400 <b>Description:</b> When the renewal quantity is below the respective MOQ offer quantity.
5125	Invalid Request for Discount Code		400 <b>Description:</b> When the customer is trying to downgrade to lesser MOQ offer than earlier.
5126	Invalid Current Quantity for Discount Code		400 <b>Description:</b> When the customer is not eligible for the requested

Error Code	Message	Applicable API calls	HTTP Status Code
			MOQ offer based on the current quantity.
5127	Discount Code is not valid for the current time period.		400 <b>Description:</b> When the customer is opting MOQ outside the opt-in window (AD-7 to AD-3 is the current window for opting)
5128	Discount code not applicable for non 3yc customer		400 <b>Description:</b> When a non-3YC customer is trying to request for MOQ offer.
5129	Discount Code not applicable for 3yc committed quantity		400 <b>Description:</b> When MPQ is less than the requested MOQ offer quantity.
5131	Invalid Order Quantity for Discount Code		400 <b>Description:</b> When the NEW or Add-on order quantity is below the

Error Code	Message	Applicable API calls	HTTP Status Code
			respective MOQ offer quantity.
5132	Discount Code not applicable for the current term		400 <b>Description:</b> When the customer is opting for same MOQ offer in the new 3YC term.
5133	Order cancellation not allowed below MOQ quantity		400 <b>Description:</b> When the customer is trying to cancel the MOQ offer quantity that results in pushing the remaining qty below the MOQ offer quantity.
5134	Could not turn off Auto Renewal for MOQ offer		400 <b>Description:</b> When the customer is trying to turn off autorenewal for an MOQ offer.
5135	Invalid Discount Code	Create Order	400



**2129 Ineligible Purchase REASON\_CODE List**

These REASON\_CODE values will be included in the additionalDetails array for 2129 errors.

REASON_CODE	Description	Action Required
INELIGIBLE_SWITCH	The order includes products that are incompatible with each other or with a product the customer already has.	Try removing the incompatible line item and placing the order again
INELIGIBLE_ADD_ON	Customer is attempting to purchase an add-on product without the base product.	Try placing the order again with the base product included
INELIGIBLE_CONSUMED	RETURN order only: customer is attempting to return a consumable product that has already been consumed.	Order cannot be returned if the transactions are consumed
INELIGIBLE_MARKET_SEGMENT	The order includes an item that belongs to a market segment that either the reseller or the customer do not belong to.	Ensure the correct offer is used for the customer and reseller's market segment.

**5117 Ineligible Transfer REASON\_CODE List**

These REASON\_CODE values will be included in the additionalDetails array for 5117 errors.

REASON_CODE	Description	Action Required
INVALID_COUNTRY	Customer is not in a valid country for this distributor	Try a distributor in a different region or wait until the distributor supports customer's country.
BAD_MARKET_SEGMENT	Customer's market segment is either not supported for this distributor or for VIPMP	Try a different distributor that supports customer's market segment or wait for a VIPMP change to allow the market segment.
INACTIVE	VIP Account is inactive	Customer/reseller must reactivate VIP account.

REASON_CODE	Description	Action Required
EXTENDED_TERM_3YC	Customer's contract term length is not supported by VIPMP	Wait for terms to be supported by VIPMP or wait until current term ends.
OPEN_PURCHASE_AUTH	Customer has open purchase authorization(s)	Open PAs must be cancelled/returned (if possible) or converted to order and paid for.  Can also be bypassed with <i>expire-open-pas</i> query parameter
CONTRACT_NOT_ACCEPTED	Customer's VIP contract has not been accepted	Customer must set up an administrator to accept VIP contract.
IN_WINDOW_PARTIAL_RENEWAL	Customer is in the renewal window prior to their anniversary date and they have partially renewed.	Customer must either renew all active licenses or wait until their anniversary date.
RETURNABLE_PURCHASE	Customer has order(s) still in the return window	Wait until return window has closed (14 days after last order).  Can also be bypassed with <i>ignore-order-return</i> query parameter.
NO_TARGET_SKU	Customer has product(s) that are not valid VIPMP product(s)	Wait for products to be supported in VIPMP.
UNBILLED_ORDER	Customer has unbilled order items	Customer must pay for all orders. If they have not received an invoice, then the customer needs to contact Adobe.

REASON_CODE	Description	Action Required
UNPAID_INVOICE	Customer has open/unpaid invoice(s)	Customer must pay for open invoice(s).
LINK_MEMBERSHIP_CHECK	Customer is part of linked membership	Wait for the Linked Membership transfer capability to be supported
HAS_UNPROCESSED_ORDERS	There is some internal issue for the customer	Please reach out to support for assistance
NOT_A_FULL_RENEWAL_IN_RENEWAL_WINDOW	The customer is in the renewal window prior to their anniversary date, and they have partially renewed.	Customer must either renew all active licenses or wait until their anniversary date.
LICENSE_WITHOUT_SALE_AQO_REFERENCED	There is some internal issue for the customer	Please reach out to support for assistance
SRC_DST_OFFER_IDS_HAS_PA_MISMATCH	There is some internal issue for the customer	Please reach out to support for assistance
3YC_LINKED_MEMBERSHIP	Customer is part of linked membership and has active 3YC	Please reach out to support for assistance
INVALID_PRICE_LEVEL_LM_MIGRATION	Customer's Price Level is greater than 04 and has Linked Membership	Please reach out to support for assistance
LM_OWNER_NOT_MIGRATED	Group member is trying to migrate to VMP prior to Group Owner migration	Wait for the Group Owner to migrate first

REASON_CODE	Description	Action Required
NOT_A_WORLDWIDE_PURCHASER	Customer is WorldWide, but Parnter is not tagged as WorldWide	Please reach out to support for assistance
3YC_DISCOUNT_LEVEL_MISMATCH	Customer's discount level is greater than the level determined by MCQ	Please reach out to support for assistance
HVD_CUSTOMER	Customer is not eligible for transfer.	Please reach out to support for assistance
MINIMUM_HVD_QTY_NOT_MET	Customer is not eligible for transfer.	Please reach out to support for assistance

### 33. Supported Countries and Locales

#### Supported/Unsupported Countries

All [ISO 3166-1 2-character country codes](#) are supported, except for the following table. For the most up to date list of supported countries and their postal code regexes, please visit the [Partner Sandbox Portal](#).

ISO Code	Country Name	Unsupported Reason/Remediation
AQ	Antarctica	No population
AX	Åland Islands	Use country FI with region 01
BL	Saint Barthélemy	Use country FR with region BL
BQ	BES Isalnds	Use country NL with region BQ1, BQ2, or BQ3
BV	Bouvet Island	No population
CU	Cuba	Embargoed
CW	Curacao	Use country NL with region CW
EH	Western Sahara	
GG	Guernsey	Use country GB
GS	South Georgia and The South Sandwich Islands	
HM	Heard Island and McDonald Islands	No population
IM	Isle of Man	Use country GB
IO	British Indian Ocean Territory	Use country GB
IR	Iran	Embargoed
JE	Jersey	Use country GB

ISO Code	Country Name	Unsupported Reason/Remediation
KM	Comoros	
KP	North Korea	Embargoed
MF	Saint Martin (French part)	Use country FR
PN	Pitcairn	
PW	Republic of Palau	
SD	Sudan	Embargoed
SJ	Svalbard and Jan Mayen Islands	
SS	South Sudan	
SY	Syria	Embargoed
TF	French Southern Territories	Use country FR

Supported Locales (Preferred Language)

The following table has a list of all supported locales for the **preferredLanguage** field of Reseller and Customer Account creation.

Locale
bg-BG
cs-CZ
da-DK
de-DE
en-AE

Locale
en-GB
en-IL
en-US
en-XM
es-ES
es-LA
es-MX
es-NA
et-EE
fi-FI
fr-CA
fr-FR
fr-MA
fr-XM
hr-HR
hu-HU
it-IT
ja-JP
ko-KR
lt-LT
lv-LV

Locale
nb-NO
nl-NL
pl-PL
pt-BR
ro-RO
ru-RU
sk-SK
sl-SI
sr-RS
sr-SP
sv-SE
tr-TR
uk-UA
zh-CN
zh-HANS
zh-HANT
zh-TW

34. Resources and Fields

Please see the [Validations/Regular Expressions](#) section for any regular expressions for specific fields.

Reseller (top-level resource)

Property	Type	Description	Range/Limits
externalReferenceId	string Optional	Marketplace’s ID for reseller. ID does not need to be unique.	Max: 35 characters



distributorId	string	Adobe-generated unique ID for distributor tied to this reseller	Max: 40 characters
resellerId (read only)	string	Unique ID for reseller created upon account creation	Max: 40 characters
companyProfile	<b>CompanyProfile</b> resource	Information about the reseller	
creationDate (read only)	String (datetime)	Date and time of account creation in UTC	
status (read only)	string	Status code for the reseller account	4 characters
links (read only)	<b>Links</b> resource	Deep links to get customer account details	

**Customer** (top-level resource)

Property	Type	Description	Range/Limits
externalReferenceId	string <b>Optional</b>	Marketplace's ID for customer. ID does not need to be unique.	Max: 35 characters
customerId (read only)	string	Unique ID for customer created upon account creation	Max: 40 characters
resellerId	string	ID of reseller tied to customer	Max: 40 characters
globalSalesEnabled	String	Global status of a customer	
linkedMembership	String	Information about the linked membership	
companyProfile	<b>CompanyProfile</b> resource	Information about the customer	
cotermDate (read only)	String (date)	Date that renewal order is to be placed. Should be one year after first order is provisioned (if a 1-yr term) and gets updated upon each renewal order.	10 characters
creationDate (read only)	String (datetime)	Date and time of account creation in UTC	
status (read only)	string	Status code of customer account	4 characters
links (read only)	<b>Links</b> resource	Deep links to get customer account details	

**Deployment** (top-level resource)

Property	Type	Description	Range/Limits
deploymentId	String	Unique ID for the deployment generated upon deployment creation	Max: 40 characters
companyProfile	<b>CompanyProfile</b> resource	Information about the customer	
creationDate (read only)	String (datetime)	Date and time of account creation in UTC	
status	string	Status code of customer account	4 characters
links (read only)	<b>Links</b> resource	Deep links to get customer account details	

## CompanyProfile

Property	Type	Description	Range/Limits
companyName	string	Name of company	Min: 4 characters Max: 80 characters
preferredLanguage	string	Preferred language for company (IETF hyphenated language tag)	Max: 40 characters
address	<b>Address</b> resource	Address of company	
contacts	Array of <b>Contact</b> resources	Points of contact for admin console	Min: One contact No limit

## Address

Property	Type	Description	Range/Limits
country	string	Country of address (ISO 3166 standard)	2 or 3 characters
region	string	State/region of the address (ISO 3166-2 standard – only send code after hyphen). Ex. US-CA → CA	Max: 255 characters
city	string	City	Max: 40 characters
addressLine1	string	Line 1 of the address	Max: 60 characters
addressLine2	string	<b>Optional.</b> Line 2 of the address	Max: 60 characters
postalCode	string	Zip code/postal code of address. Must match city/region/country.	Max: 40 characters
phoneNumber	string	<b>Optional.</b> Phone number associated with this address	Max: 40 characters

## Contact

Property	Type	Description	Range/Limits
email	string	Email address of the contact	Max: 240 characters
firstName	string	First name of the contact	Max: 35 characters
lastName	string	Last name of the contact	Max: 35 characters
phoneNumber	string	<b>Optional.</b> Phone number of the contact	Max: 40 characters

**Order** (top-level resource)

Property	Type	Description	Range/Limits
externalReferenceId	string <b>Optional</b>	Reconciliation ID for this order (ex. PO Number, Order ID, etc.). ID does not need to be unique.	Max: 35 characters
orderId (read only)	string	Unique identifier for order that is created upon order creation	Max: 40 characters
customerId (read only)	string	ID of the customer that placed this order	Max: 40 characters
referenceOrderId	string <b>required for RETURN orders</b>	The ID of the order being returned in a RETURN order.	Max: 40 characters
orderType	String (enum)	Type of the order. Valid values: NEW, PREVIEW, RETURN, TRANSFER, RENEWAL	
lineItems	Array of <b>LineItem</b> resources	Itemized list of the offers and their quantity for this order	Min: 1 item Max: 499 items
currencyCode	string	Currency used for placing the order (ISO 4217 format).	3 characters
creationDate (read only)	String (datetime)	Date and time the order was created in UTC	
status (read only, except for cancelling)	string	Status code of the order	4 characters
links (read only)	<b>Links</b> resource	Deep links to get order details	

**LineItem**

Property	Type	Description	Range/Limits
extLineItemNumber	integer	Marketplace's unique index for item in order	Min: 0 Max: 999999 <b>Must be unique</b>
offerId	string	The ID of the offer being purchased	
quantity	integer	Quantity of this offer purchased in this order	Min: 1 Max: 10,000 for Team and 599,000 for Enterprise
discountCode	string	The discount code applicable to the HVD customers migrating from VIP to VIP Marketplace.	Max: 40 characters
subscriptionId (read only)	string	ID of the Subscription resource associated with this line item.	Max: 40 characters
status (read only)	string	Status code of the line item	4 characters

**Eligible Offers**

Property	Type	Description	Range/Limits
offerId	string	The unique ID of the offer.	Max: 40 characters
renewalCode	string	Unique identifier of the Minimum Order Quantity Offer. Available values are: <ul style="list-style-type: none"> <li>MOQ_100</li> <li>MOQ_250</li> <li>MOQ_500</li> </ul>	Max: 30 characters
eligibilityCriteria	Array	The eligibility criteria for availing the MOQ offer.	
minQuantity	Integer	The minimum quantity for which this offer is applicable, also the minimum quantity that the customer needs to commit for a 3YC term to be eligible for this offer.	Min: 0 Max: 999999
additionalCriteria	string	The additional criteria list for availing the High Growth Offer. Currently, THREE_YEAR_COMMIT is the only supported value, indicating that 3YC is required to avail the High Growth Offers.	Min: 1 item Max: 499 items
deploymentId	string	Unique ID of the deployment.	Max: 40 characters

**Subscription** (top-level resource)

Property	Type	Description	Range/Limits
subscriptionId (read only)	string	Unique identifier of the subscription resource	Max: 40 characters
currentQuantity (read only)	integer	Total quantity of the subscription that a customer has	Max: 10,000 for Team / 599,000 for Enterprise
usedQuantity	integer	Total used quantity for the license or the consumable	
offerId	string	The base discount level offer id. <b>Note:</b> This attribute is included for developers to identify the offer and NOT the discount level. Partners need to use “Get customer Account” and “Get Order history” APIs to see discount level information.	
currencyCode	string	Currency used for placing the subscription (ISO 4217 format). Mandatory for Global customers.	3 characters
deploymentId	String	Unique ID for the deployment generated upon deployment creation. Applicable for global customers.	Max: 40 characters
autoRenewal	<b>AutoRenewal</b> object	Resource to manage auto-renewal flag and quantity	
allowedActions	string	Indicates the actions that can be performed on a subscription. The value "MANUAL_RENEWAL" indicates that the subscription can be selected for manual renewal.	
creationDate (read only)	String (datetime)	Date and time of subscription creation in UTC	

status (read only)	string	Current status code of the subscription	4 characters
links (read only)	<b>Links</b> resource	Deep links to get subscription details	

**Notification** (top-level resource)

Property	Type	Description	Range/Limits
limit	integer	Maximum number of notification items response can contain	0 to INTEGER_MAX
offset	integer	Number of notification items to skip before picking up the items	0 to INTEGER_MAX
totalCount	integer	Total number of items present for a notification	0 to INTEGER_MAX
count	integer	Actual number of notification item response contains	0 to INTEGER_MAX
items	Array of <b>NotificationItem</b> resources	Itemized list of notifications	0 to INTEGER_MAX
links(read only)	<b>Links</b> resource	Deep links to get notification details	

**NotificationItem**

Property	Type	Description	Range/Limits
customerId	string	Customer id for the account where partner order is pending	Max: 40 characters
resellerId	string	Reseller account id for the account whose partner order is still pending	Max: 40 characters
notificationType	string	Type of notification response	Max: 40 characters
links	<b>Links</b> resource	Deep links to get notification item details	

**AutoRenewal**

Property	Type	Description	Range/Limits
enabled	boolean	Boolean flag denoting whether or not autoRenewal is enabled	true or false

renewalQuantity	integer	<b>Optional.</b> Total quantity to be renewed at the end of the contract period (can be higher than the current subscription quantity)	Min: 1 Max: 10,000 for Team / 599,000 for Enterprise
-----------------	---------	--	---

Links

Property	Type	Description	Range/Limits
self	<b>Link</b> object	Link to get data about this resource	
next	<b>Link</b> object	Link to get data about the next resource	
prev	<b>Link</b> object	Link to get data about the previous resource	

Link

Property	Type	Description	Range/Limits
uri	string	The URI to access this link	
method	string	The method used for the URI	
headers	Array of Key:Value pairs	The headers for the link. <b>Note:</b> Any headers in this are in addition to the required headers specified in the Headers section above.	

## 35. Validations/Regular Expressions

Field Name	Resource/Object	Regular Expression (Java String)	Notes
companyName	CompanyProfile	<code>^[\p{L}\p{N}_ ,.&amp;&amp;'\'() \\"\\/-]{4,80}\$</code>	Full Unicode letter and number support – no normalization is done.
firstName	Contact	<code>^[\p{L}\p{N}-_ ,.&amp;&amp;'\'"]{1,35}\$</code>	Full Unicode letter and number support – no normalization is done.
lastName	Contact	<code>^[\p{L}\p{N}-_ ,.&amp;&amp;'\'"]{1,35}\$</code>	Full Unicode letter and number support – no normalization is done.
email	Contact	<code>(?:[a-zA-Z0-9!#\$%&amp;'*/+=?^_`{ }~-]+(?:\\. [a-zA-Z0-9!#\$%&amp;'*/+=?^_`{ }~-]+)* "(?:[\x01-\x08\\x0b\\x0c\\x0e-\\x1f\\x21\\x23-\\x5b\\x5d-\\x7f] \\\\"[\x01-\\x09\\x0b\\x0c\\x0e-\\x7f])*")@(?:(?:[a-zA-Z0-9](?:[a-zA-Z0-9-]*[a-zA-Z0-9])?\\. [a-zA-Z0-9](?:[a-zA-Z0-9-]*[a-zA-Z0-9])?) \\[(?:(?:(25[0-5] 2[0-4][0-9] [01]?[0-9][0-9]?)\\.){3}(?:25[0-5] 2[0-4][0-9] [01]?[0-9][0-9]?) \\[a-zA-Z0-9-]*[a-zA-Z0-9]: (?:[\\x01-\\x08\\x0b\\x0c\\x0e-\\x1f\\x21-\\x5a\\x53-\\x7f] \\\\"[\x01-\\x09\\x0b\\x0c\\x0e-\\x7f])+)\\])</code>	Source: <a href="https://emailregex.com">emailregex.com</a>
postalCode	Address	Varies based on country. Please see the <a href="#">Supported Countries and Locales</a> section	

## 36. Support

### Contact Support

- For development and integration support, please contact [partnerapi-integration-support@adobe.com](mailto:partnerapi-integration-support@adobe.com)
- For production API support, please contact [partnerapi-support@adobe.com](mailto:partnerapi-support@adobe.com)

### Reference Links

- Additional details about IMS [access tokens](#)
- Additional details about the [JWT authentication workflow](#)
- Explore more tutorials provided by Adobe I/O [here](#)
- JWT libraries and SDKs can be found [here](#)