https://cloudblue.com

Subscription Change

This article has been generated from the online version of the documentation and might be out of date. Please, make sure to always refer to the online version of the documentation for the up-to-date information.

Auto-generated at November 1, 2026

© Ingram Micro, Inc. All Rights Reserved.

Overview

When customers change their subscriptions, new **change** fulfillment requests are generated on the CloudBlue Connect® platform. Note that such fulfillment requests refer to changing the subscribed Items (SKUs) or adjusting the Item quantity. Once a change request is approved, its corresponding subscription is updated by Connect.

The initial status of a change fulfillment request is *Pending*. It remains in this state until it is processed via the Connect platform or via your deployed middleware. In case a change request is successfully processed, its status is changed to *Approved*.

Vendors can deploy the Processor SDK to approve pending change fulfillment request by calling the Vendor API. Therefore, Vendors can change or update active subscriptions via their systems.

Prerequisites

Before following the provided use case, make sure that all of these prerequisites are met:

- Your product is defined on the Connect platform.
- An active subscription is required to create a change request.
- The Processor SDK Template is successfully deployed.
- Your deployed Processor is properly configured.
- Vendor API configuration is presented.

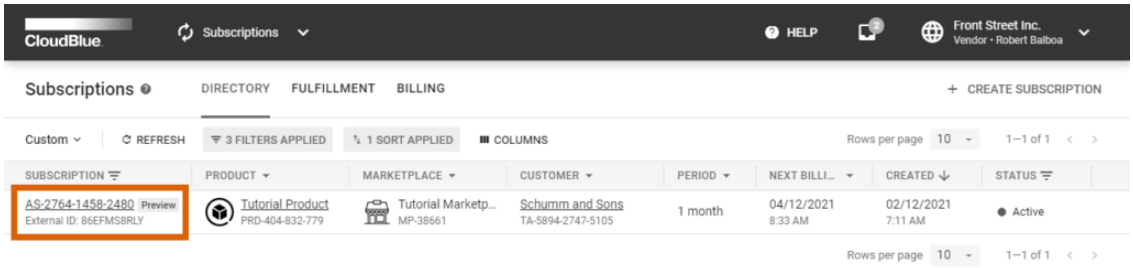
Creating Change Requests

Change fulfillment requests for active subscriptions are created by customers. However, Vendors can can also create this request by accessing an active preview subscription. Follow the steps below to create a change request for your generated preview subscription.

1. Access active subscription details

Click on your active preview subscription from the **Subscriptions** module of the Connect platform.

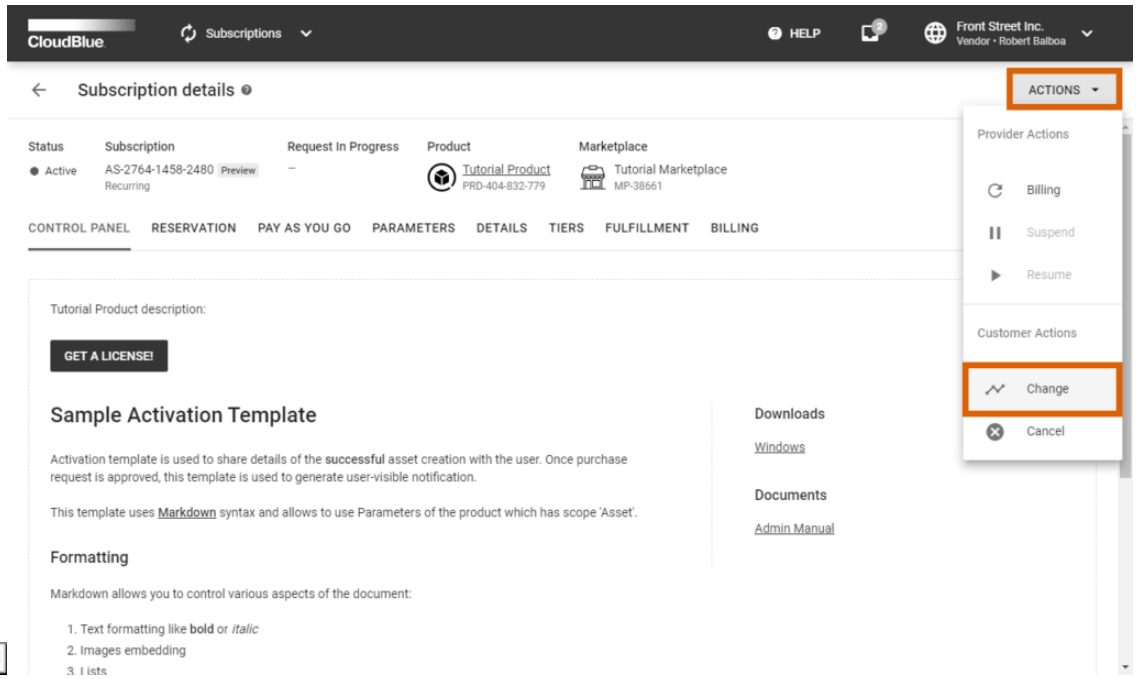
© Ingram Micro, Inc. All Rights Reserved.



Your subscriptions are displayed within the **Directory** tab. Make sure a subscription that you want to change is in the *Active* state and it is marked with the **Preview** tag.

2. Launch a Change Request Wizard

Open the **Actions** menu from the subscription details screen and click the **Change** button.

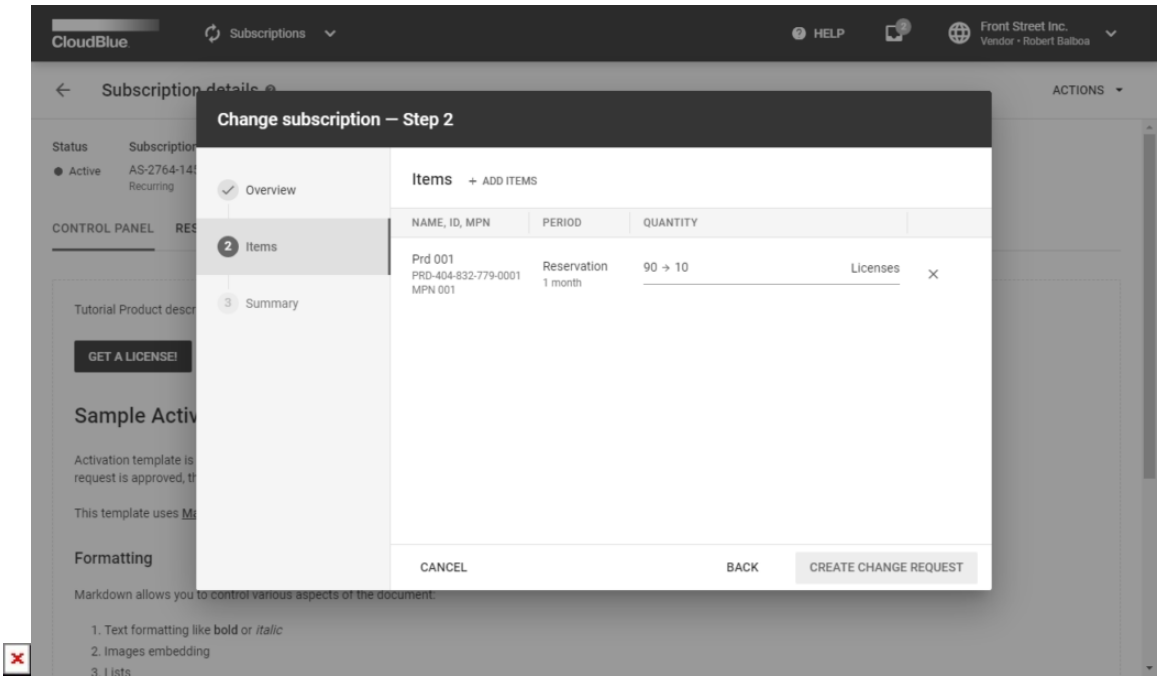


Follow the wizard steps to successfully create a change request.

© Ingram Micro, Inc. All Rights Reserved.

3. Configure your Change Request

Add your Items, change them, or adjust their quantity by using corresponding buttons from the user interface.



Once a Vendor clicks the **Create Change Request** button, the Connect platform generates a fulfillment request to change a subscription.

Code Snippets

Create a script that triggers specific API requests to approve the aforementioned change requests. Thus, subscriptions can be updated or changed by approving change requests via your configured Processor SDK Template.

> ●
> Item Quantity Reduction
>
> In case your business does not support and does not allow reducing the subscribed items quantity, it is necessary to fail the request with a corresponding message.

```
from connect_processor.app.utils.utils import Utils
from cnct import ConnectClient
from typing import Dict


class Change:
```


© Ingram Micro, Inc. All Rights Reserved.

```
# This class is used to change items or their number in an active subscription.


 @staticmethod
 def process_request(request, client):
# type: (Dict, ConnectClient) -> Dict
# This method processes fulfillment requests in Pending status.
# In case the Vendor system does not support subscription changes,  check the request details and reject
the request.

 # Get the subscription ID saved as a fulfilment parameter to prepare the body/payload for the Vendor API.
 # The Subscription ID can be specified as a fulfillment parameter ID.
 # The external_subscription_id is used to call the Vendor API and change the subscription.
 external_subscription_id = Utils.get_param_value(request, 'fulfillment', 'subscription_id')

# Add the code to update or change the subscription in the vendor system by calling the Vendor API.
 # api_client = APIClient(api_url='',
 # api_key='')
 # change_payload = {}
 # api_client.change_subscription(change_payload, external_subscription_id)

 # Approve the change request with the following code:
 return Utils.approve_fulfillment_request(request, client)
# In case all changes are applied successfully, the subscription status remains Active.
 # If the changes are not applied, the request can be rejected by using the
Utils.reject_fulfillment_request method.
```

Summary

As a result, the Processor helps Vendors approve change requests on Connect from their systems. The Connect platform will also update subscriptions with approved change requests.

In case changes in the subscription cannot be applied, the Processor also helps Vendors reject these change requests.

© Ingram Micro, Inc. All Rights Reserved.