**BM Extension to update the allocation of products**

Graphical user interface, text, application, email

Description automatically generated

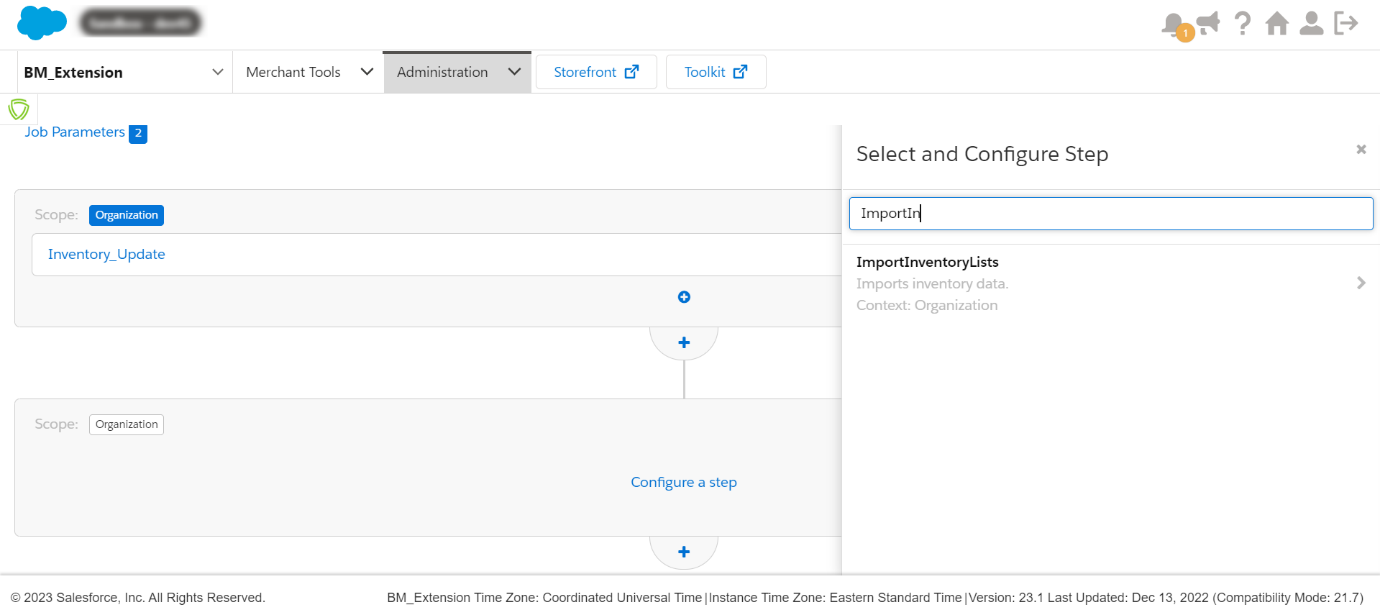
This BM (Business Manager) extension plug-in cartridge is responsible for bulk updating the allocation of products.

**How to use:**

1. First enter the inventory id where you want to change the inventory of the product.
2. Enter product ids and their corresponding allocation values.
3. Choose the corresponding perpetual checkbox if you want the allocation of the product to be infinite.
4. Use Add(+) button to add multiple product ids.
5. Enter your e-mail id where you want to get the result of the allocation process.

**Dependencies:**

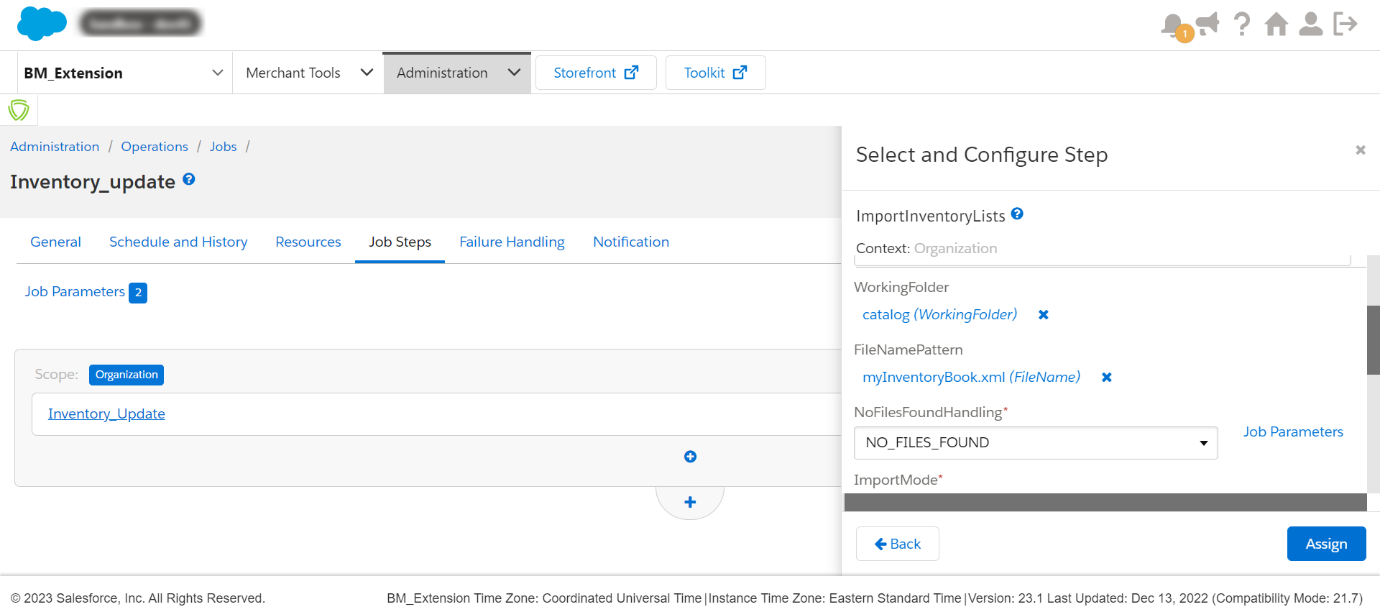
1. **Job Dependencies:**
   1. Create a job with the id **Inventory\_update**, inside the job create a job step with the id **Inventory\_Update** and select **ImportInventoryLists.**



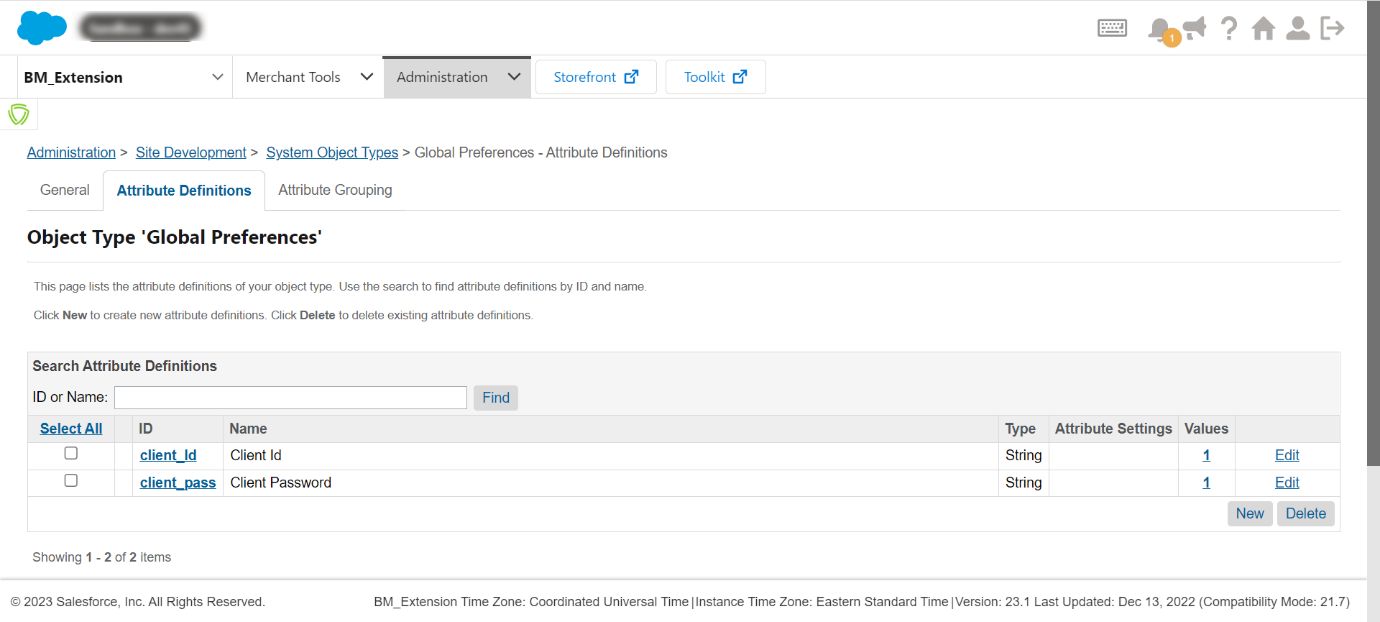
* 1. Set the import mode to Merge, the working folder to catalog, and set the name of the file to **myInventoryBook**.**xml,** and leave the rest of the parameters on default.

Graphical user interface, application

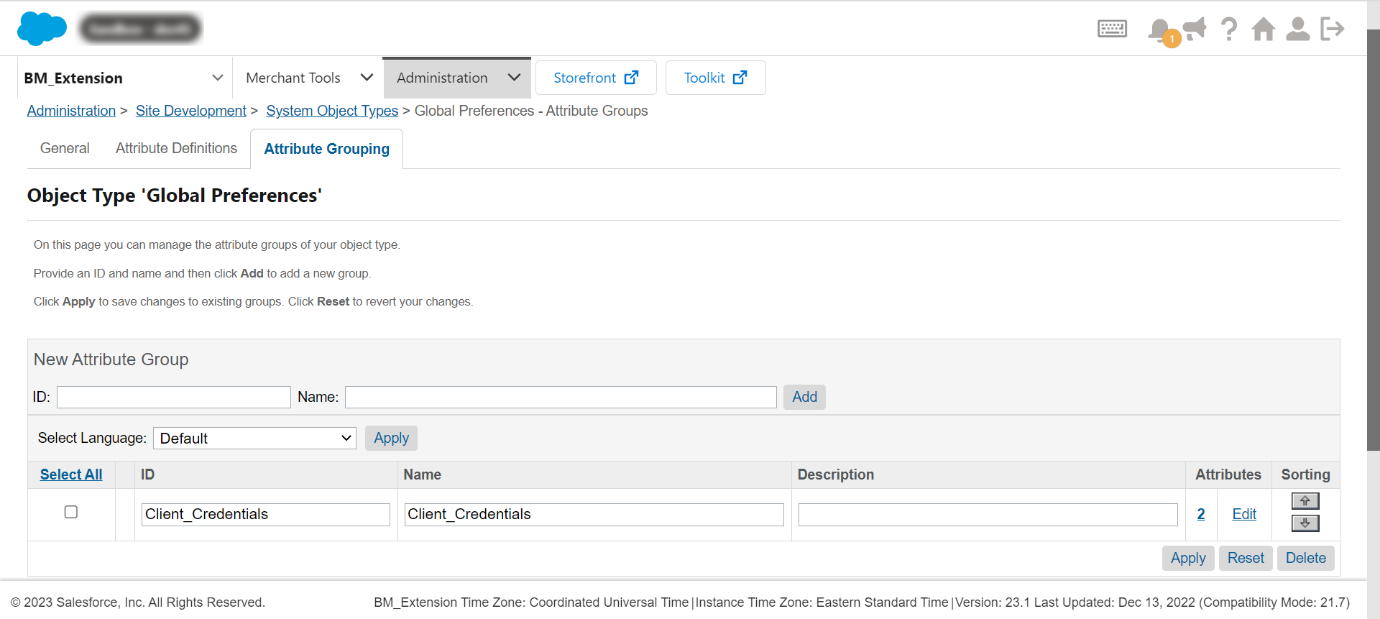
Description automatically generated

****

1. **Global Preferences**
   1. Go to Administration>Site Development>System Object Types> OrganizationPreferences(Global Preferences).
   2. Create two attribute definitions, first with the id client\_Id and name Client Id of type string, second with the id client\_pass and name Client Password of type string.



* 1. Now create an Attribute grouping with id Client\_Credentials and name Client\_Credentials and assign the above-created attribute definitions to the group.
  2. Now you will be able to see this customer group in Administration> Global Preferences> Global Custom preferences> Client\_Credentials. Enter your Client Id and Client password inside the fields.



1. **Service Dependencies**

For this functionality to run, we need three services. One service will fetch the BM (Business Manager) access token so that we can call other OCAPIs that need an access token. The second service is for the OCAPI which will call the **Job** that we have configured in the previous step. The last service is to check the status of the second service if the second service response is okay then it will send an acknowledgment to the entered email-id.

* 1. Create a service with id Create\_token, go to Service Credentials create a credential with the id Create\_token put the url to

<https://account.demandware.com/dw/oauth2/access_token>? and put the

user( the user id in ocapi data json) id to the User field and set the password.

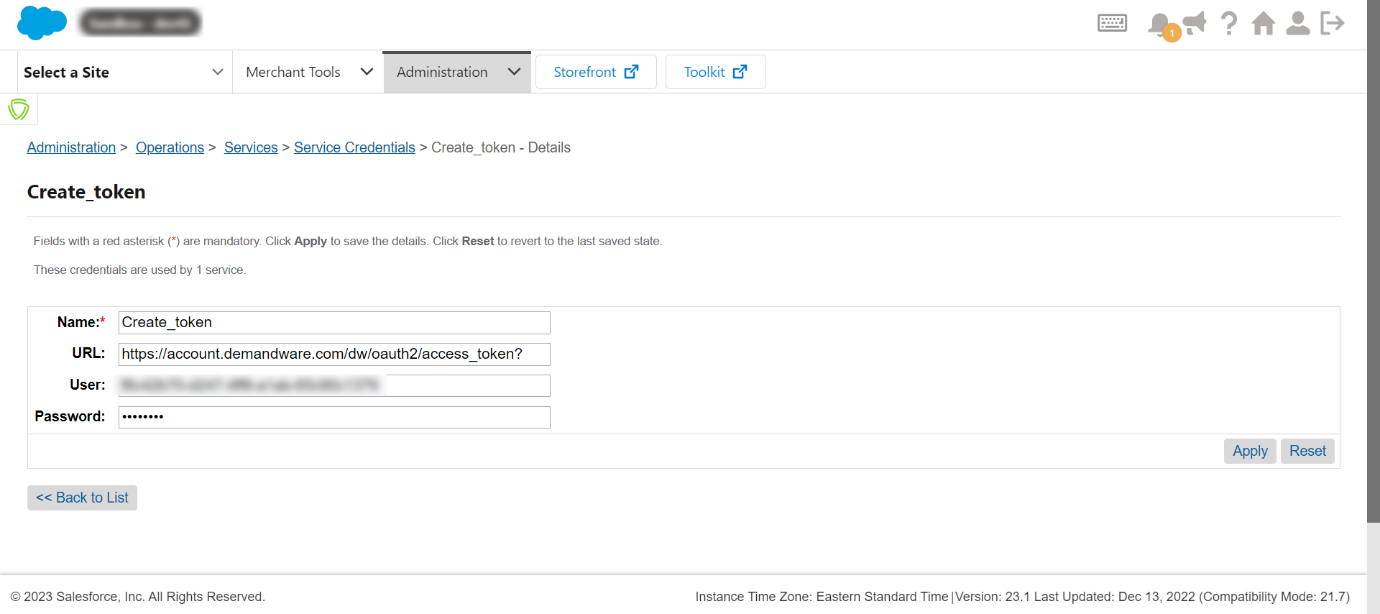
Now create the service profile with the Name Create\_token set Enable circuit

Beaker to tick. Now go to your service and bind the service profile and service

The credential you have just created with the id Create\_token. Set the service mode

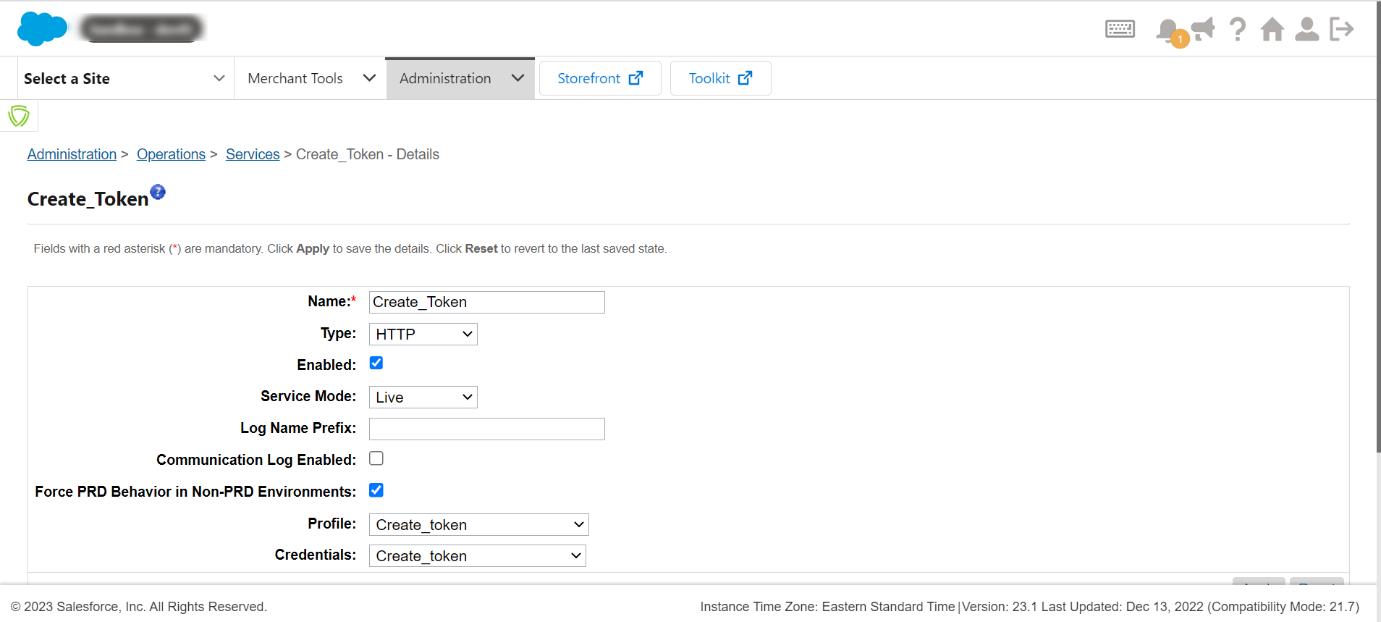
To Live and tick the Enabled checkbox.

Tip: Give the same name to the credential and profile as of the service to avoid any ambiguity.



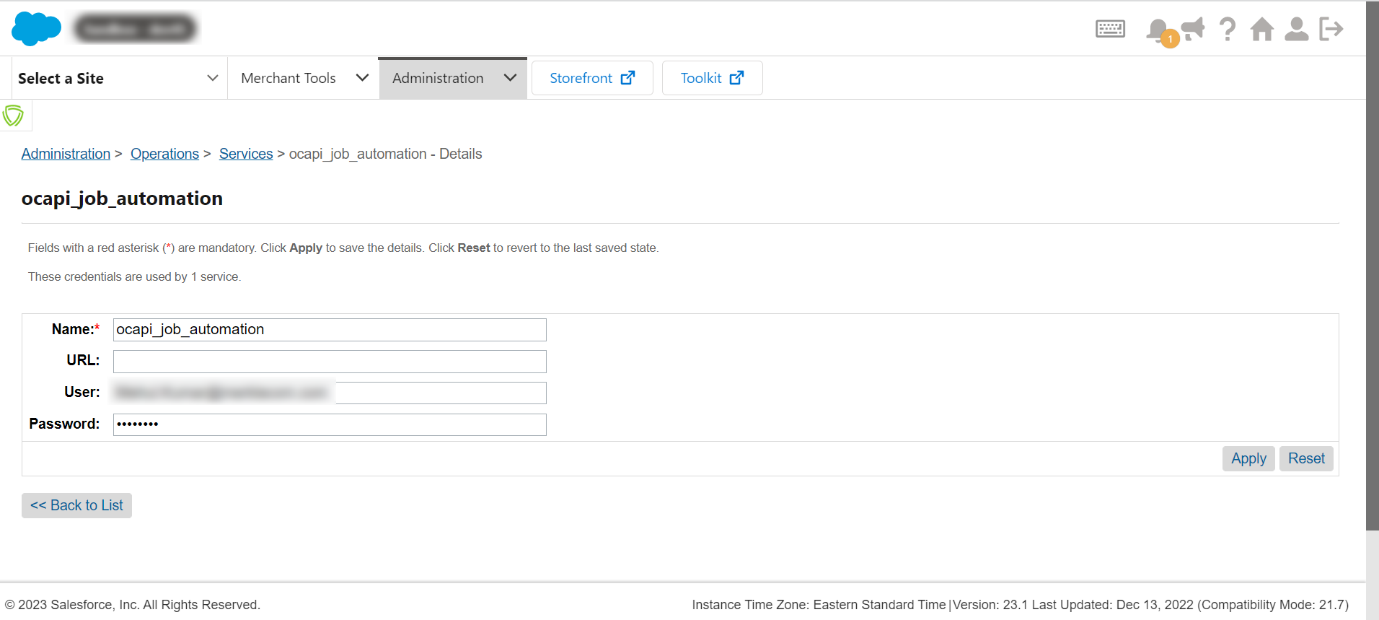
**Graphical user interface, application

Description automatically generated**

****

* 1. Create a service with the id **ocapi\_job\_automation,** go to Service Credentials create a credential with the id **ocapi\_job\_automation** leave the URL empty as the url is constructed in the code itself, and put the user(the user id in ocapi data json) id to the User field and set the password.

Now create the service profile with the Name **ocapi\_job\_automation** set Enable circuit Beaker to tick. Now go to your service and bind the service profile and the credential you have just created with the id **ocapi\_job\_automation**. Set the service mode to Live and tick the Enabled checkbox.



Graphical user interface, text

Description automatically generated

* 1. Create a service with the id **ocapi\_job\_status**, go to Service Credentials create a credential with the id **ocapi\_job\_status** leave the URL empty as the url is constructed in the code itself, and put the user( the user id in ocapi data json) id to the User field and set the password.

Now create the service profile with the Name **ocapi\_job\_status** set Enable circuit Beaker to tick. Now go to your service and bind the service profile and the credential you have just created with the id **ocapi\_job\_status**. Set the service mode to Live and tick the Enabled checkbox.

Graphical user interface, text, application, email

Description automatically generated

**Graphical user interface, text, application, email

Description automatically generated**

**------END----**