



INVENTORY SEGMENTATION & DOM ROUTING PACKAGE TECHNICAL IMPLEMENTATION

TABLE OF CONTENTS

Prerequisites	03
Installation	03
Configuration	04
Location Group Priority Configurations	07
Extension Points	08
Segmentation Settings	08
Upload Inventory Service	15
Permission Sets	18

Prerequisites:

Make sure that you've met the prerequisites before implementing order management.

- Make sure that Lightning Experience is enabled. Order Management isn't supported on Salesforce Classic or mobile. You can view record pages on mobile but can't use order management features.
- Purchase and activate an Order Management license. A license is included with a developer org or when you're using a partner-approved Order Management template ID to create a trial org

Enable list custom setting type:

- Setup -> Data -> Schema Settings -> Manage list custom settings type (Activate)

Enable order management Doc:

https://resources.docs.salesforce.com/latest/latest/en-us/sfdc/pdf/order_management_developer_guide_html.pdf

Enable OCI if you are planning to use Omni Channel Inventory:

https://resources.docs.salesforce.com/232/latest/en-us/sfdc/pdf/salesforce_omnichannel_inventory_implementation_guide.pdf

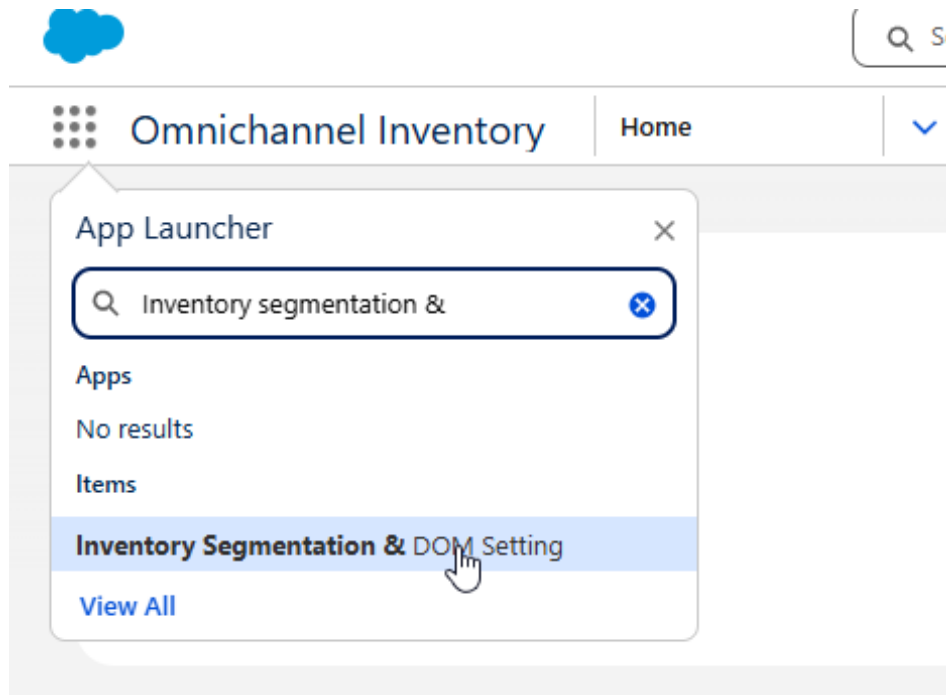
Installation

Install the package using the link:

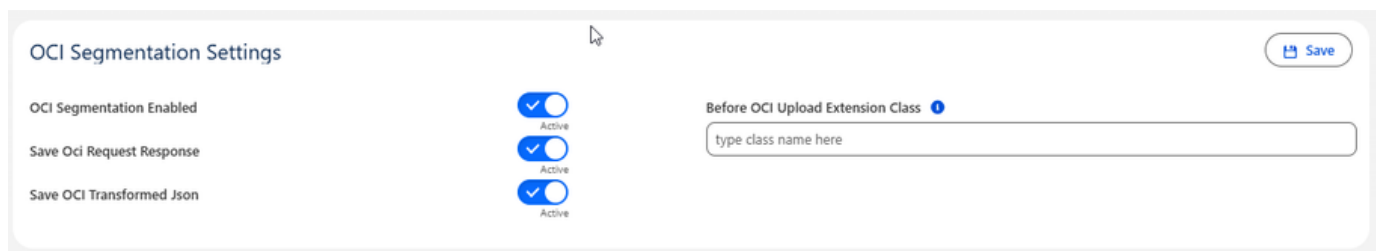
<https://login.salesforce.com/package/installPackage.apexp?p0=04tWj000000lDeXIAU>

Configuration

App Launcher -> Location Segmentation & DOM Settings

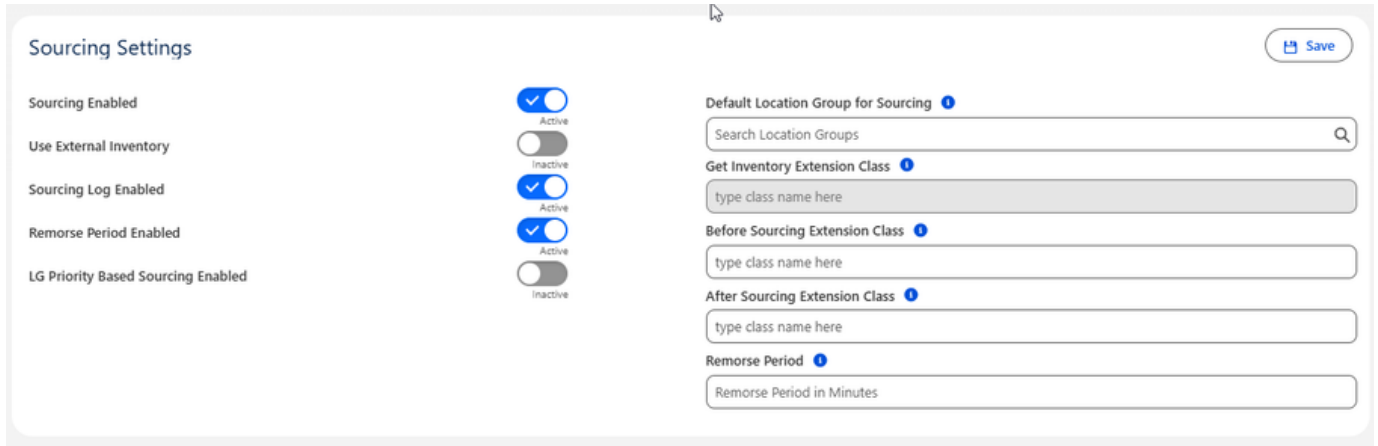


Enable OCI Segmentations

The screenshot shows the 'OCI Segmentation Settings' configuration page. On the left, there are three settings, each with a toggle switch and the word 'Active' below it: 'OCI Segmentation Enabled', 'Save Oci Request Response', and 'Save OCI Transformed Json'. On the right, there is a section titled 'Before OCI Upload Extension Class' with a blue information icon. Below this title is a text input field with the placeholder text 'type class name here'. A 'Save' button is located in the top right corner of the settings panel.

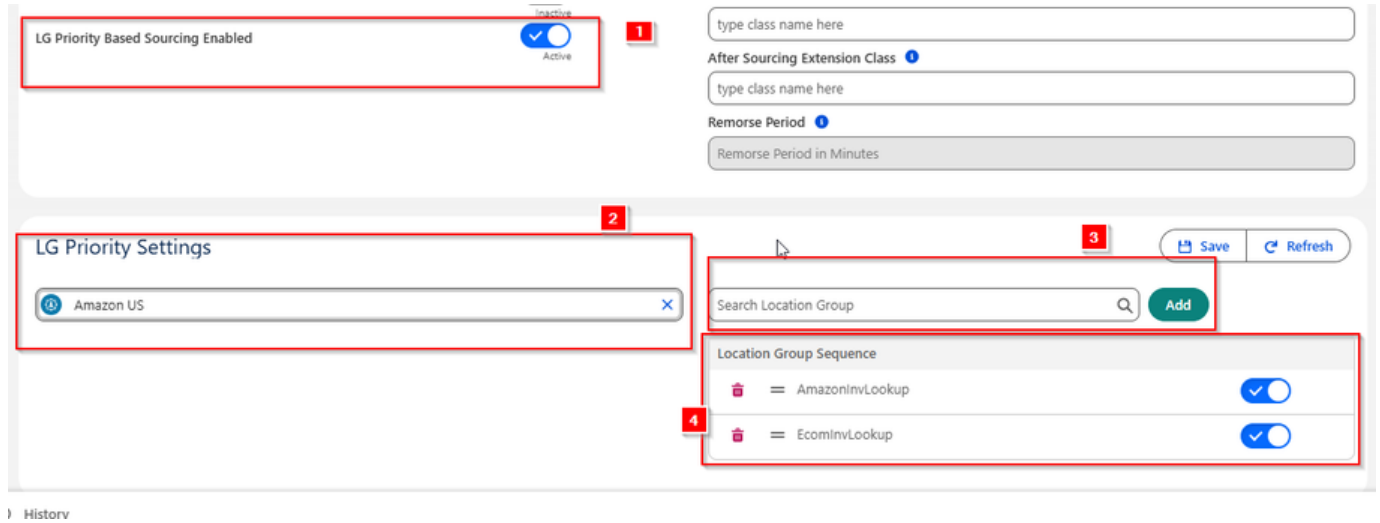
- **OCI Segmentation Enabled:** Enables the segmentation feature
- **Save OCI Request Response:** Stores the request and response provided to the REST API
- **Save OCI Transformed Json:** Stores the transformed JSON/request sent to OCI after segmentation logic
- **Before OCI Upload Extension Class:** This is the extension point for any customizations. A custom class can be created and configured here.

Enable Sourcing Settings



- **Sourcing Enabled:** Enable Sourcing feature
- **Use External Inventory:** Enable this if you want to use external inventory, not OCI
- **Sourcing Log Enabled:** Logs the sourcing decisions in the feed item of order summary.
- **Remorse Period Enabled:** Enables the remorse period for the order
- **LG Priority-Based Sourcing Enabled:** By enabling this, we can configure multiple location groups for a sales channel and its priority. This configuration will be used in the sourcing logic.
- **Default Location Group for Sourcing:** This is optional. When a sales channel is not configured with any location group to fulfill the order, this location group will be used to fulfill.
- **Get Inventory Extension Class:** This should be configured when "Use External Inventory" is enabled. This is the class to invoke for external inventory lookup.
- **Before Sourcing Extension Class:** This class will be invoked before the core sourcing logic is invoked. Any business customization logic can be implemented using this extension point.
- **After Sourcing Extension Class:** This class will be invoked after the sourcing logic is run (before creating the fulfillment order) and the fulfillment location is identified. Any custom override logic can be implemented here.
- **Remorse Period:** This should be configured when "Remorse Period Enabled" is enabled. The order waits the configured minutes before sourcing.

Location Group priority configurations:



The screenshot shows the 'LG Priority Based Sourcing Enabled' toggle (1) set to 'Active'. Below it, the 'LG Priority Settings' section (2) shows 'Amazon US' selected. To the right, the 'After Sourcing Extension Class' field is empty, and the 'Remorse Period' is set to 'Remorse Period in Minutes'. The 'Location Group Sequence' table (4) lists 'AmazonInvLookup' and 'EcomInvLookup' with toggle switches. A search bar (3) is also present.

Location Group Sequence	
AmazonInvLookup	☒
EcomInvLookup	☒

1. Enable "LG Priority Based Sourcing Enabled" toggle to setup the priority.
2. Choose the sales channel to configure the location group priority
3. Choose the Location group to add to the priority
4. Re-arrange, delete, enable/disable the location group from the priority configuration.

Extension Points

Before OCI Upload Extension Class

Interface

The class must implement the interface.

```
global interface IOCIBeforeUploadInventoryExtension {  
    void execute(List<perfawarellc.OCISegmentInput> segmentRecords);  
}
```

Input

OCISegmentInput

Field	Datatype	Description
Sku	String	
Location	String	
ociRecords	List<perfawarellc.OCIRecord>	

OCIRecord

Field	Datatype	Description
recordId	String	
sku	String	
locationId	String	
effectiveDate	DateTime	
onHand	Double	
safetyStockCount	Double	
Futures	List<perfawarellc.OCIRecord.OCIFutureRecord>	

OCIRecord.OCIFutureRecord

Field	Datatype	Description
quantity	<i>Decimal</i>	
expectedDate	<i>Datetime</i>	

Sample Implementation:

```
global class BeforeUploadInventoryExtension implements
perfawarellc.IOCIBeforeUploadInventoryExtension{
    global void execute(List<perfawarellc.OCISegmentInput>
segmentRecords){

        for(perfawarellc.OCISegmentInput record : segmentRecords){
            for(perfawarellc.OCIRecord ociRec : record.ociRecords){
                ociRec.onHand *= 2;
            }
        }
        system.debug('output: ' + segmentRecords);
    }
}
```

Get Inventory Extension Class Interface

The class must implement the interface.

```
global interface IOMGetInventoryExtension {  
    perfawarellc.OMGetInventoryOutput  
    getInventory(perfawarellc.OMGetInventoryInput input);  
}
```

Input

OMGetInventoryInput

Field	Datatype	Description
locationList	List<String>	
skuList	List<String>	

Output

OMGetInventoryOutput

Field	Datatype	Description
locationAvailabilityList	List<OMGetInventoryLocationOutput>	

OMGetInventoryLocationOutput

Field	Datatype	Description
locationidentifier	String	
skuAvailailityList	List<OMGetInventorySkuOutput>	

OMGetInventorySkuOutput

Field	Datatype	Description
Sku	String	
Atf	Double	

Sample Implementation:

```
global class GetInventoryExtension implements
perfawarellc.IOMGetInventoryExtension{
    global perfawarellc.OMGetInventoryOutput
    getInventory(perfawarellc.OMGetInventoryInput input){
        perfawarellc.OMGetInventoryOutput output = new
perfawarellc.OMGetInventoryOutput();
        List<perfawarellc.OMGetInventoryLocationOutput> locationRepList = new
List<perfawarellc.OMGetInventoryLocationOutput>();
        for(String location : input.locationList){

            List<perfawarellc.OMGetInventorySkuOutput> invRepList = new
List<perfawarellc.OMGetInventorySkuOutput>();
            for(String sku : input.skuList){
                perfawarellc.OMGetInventorySkuOutput invRep = new
perfawarellc.OMGetInventorySkuOutput();
                invRep.sku = sku;
                invRep.atf = 100;
                invRepList.add(invRep);
            }

            perfawarellc.OMGetInventoryLocationOutput locationRep = new
perfawarellc.OMGetInventoryLocationOutput();
            locationRep.locationIdentifier = location;
            locationRep.skuAvailailityList = invRepList;
            locationRepList.add(locationRep);
        }
        output.locationAvailabilityList = locationRepList;

        return output;
    }
}
```

Before Sourcing Extension

Interface

The class must implement the interface.

```
global interface IOMBeforeSourcingExtension {  
    void execute(perfawarellc.OMBeforeSourcingInput input);  
}
```

Input

OMBeforeSourcingInput

Field	Datatype	Description
segmentedItems	List<OMSegmentedItemsInput>	

OMSegmentedItemsInput

Field	Datatype	Description
locationIdentifier	String	
locationGroupIdentifier	String	
locationGroupIdentifier	List<OMFindRoutesOrderItem>	
locationGroupIdentifier	Map<String, Integer>	

OMSegmentedItemsInput

Field	Datatype	Description
quantity	Double	
stockKeepingUnit	String	
orderLineSummaryId	Id	
excludeLocations	List<String>	

Sample Implementation:

```
global class BeforeSourcingExtension implements
perfawarellc.IOMBeforeSourcingExtension{
    global void execute(perfawarellc.OMBeforeSourcingInput input){
        for(perfawarellc.OMSegmentedItemsInput segmentItem :
input.segmentedItems){
            segmentItem.locationIdentifier = 'FloridaWH';
        }
    }
}
```

After Sourcing Extension

Interface

The class must implement the interface.

```
global interface IOMAfterSourcingExtension {
    void execute(perfawarellc.OMRouteDeliveryGroupOutput input);
}
```

Input

OMRouteDeliveryGroupOutput

Field	Datatype	Description
orderSummaryId	ID	
orderDeliveryGroupSummaryId	ID	
routingGroupList	List<OMFindRoutesLocation>	
missingItems	List<OMFindRoutesRoutingItem>	

OMRouteDeliveryGroupOutput

Field	Datatype	Description
locationIdentifier	String	
locationId	ID	
routingItems	List<OMFindRoutesRoutingItem>	

OMRouteDeliveryGroupOutput

Field	Datatype	Description
stockKeepingUnit	String	
quantity	Double	
orderLineSummaryId	String	

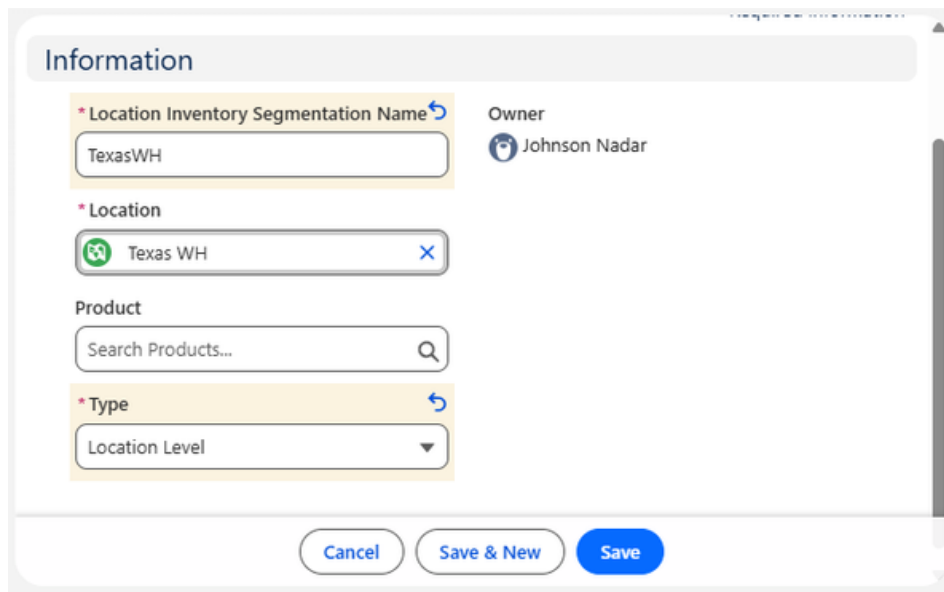
Sample Implementation:

```
global class AfterSourcingExtension implements
perfawarellc.IOMAfterSourcingExtension{
    global void execute(perfawarellc.OMRouteDeliveryGroupOutput input){
        for(perfawarellc.OMFindRoutesLocation routingGroup : input.routingGroupList){
            for(Integer i = routingGroup.routingItems.size() - 1; i >= 0; i--){
                perfawarellc.OMFindRoutesRoutingItem routingItem =
                routingGroup.routingItems[i];
                if(routingItem.stockKeepingUnit == 'Shipping'){
                    routingGroup.routingItems.remove(i);
                }
            }
        }
    }
}
```

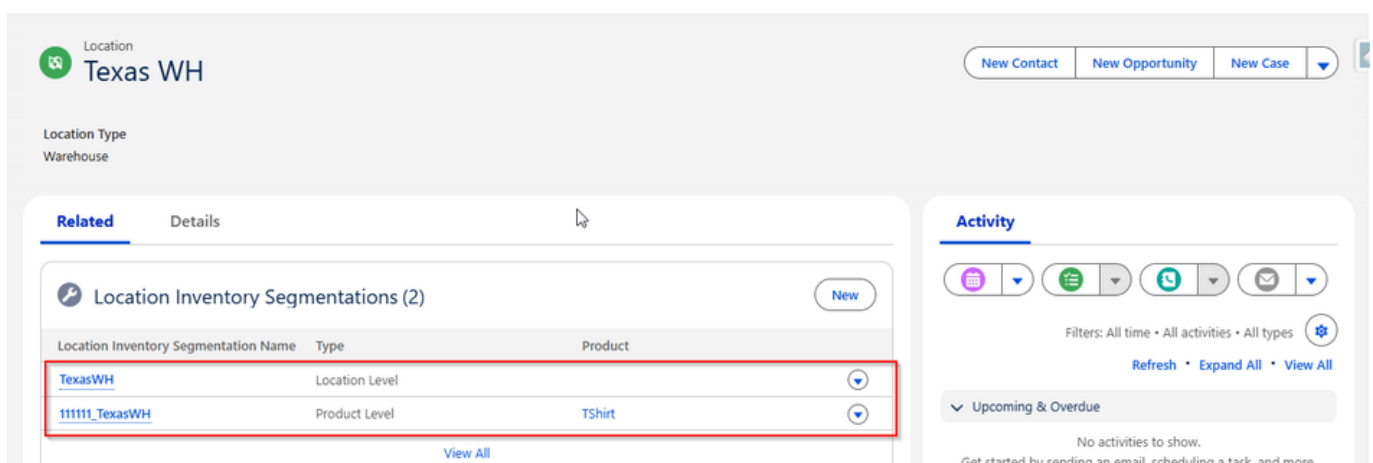
Segmentation Settings:

Segmentation Type Configuration

Go to physical Location and create the segmentation types:
Location -> Related -> Location Inventory Segmentations -> New



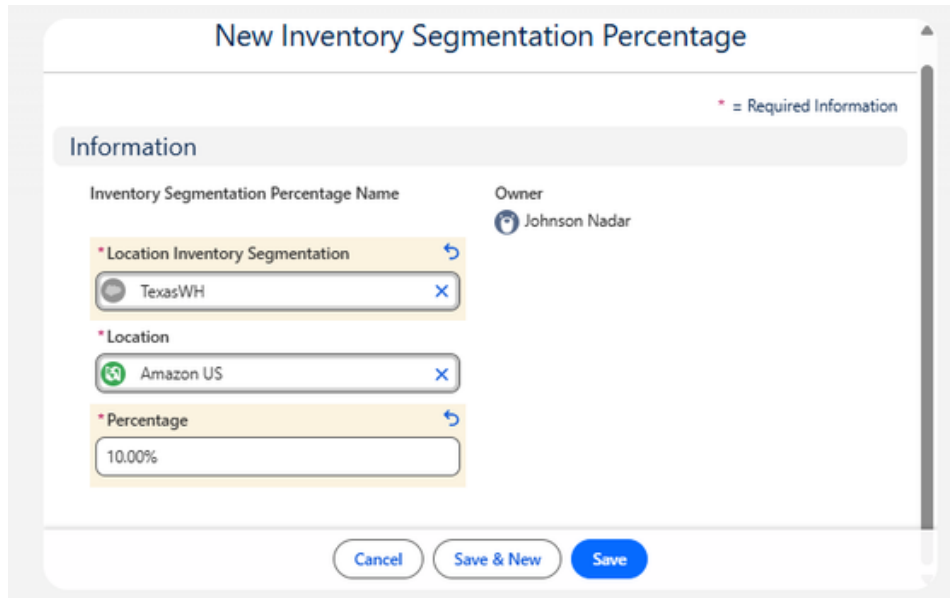
Location Inventory Segmentation Name	Provide "Test". Name of the segmentation config (this will be auto generated when you save)
Location	Physical Location
Product	Required when Type="Product Level"
Type	The segmentation Level. Valid values are "Product Level" & "Location Level"



Location Inventory Segmentation Name	Type	Product
TexasWH	Location Level	
111111_TexasWH	Product Level	TShirt

Segment percentage configuration

Configure the segment percentage in the Logical/Virtual Location:
Location -> Related -> Inventory Segmentation Percentages -> New



New Inventory Segmentation Percentage

* = Required Information

Information

Inventory Segmentation Percentage Name: (Location Inventory Segmentation)

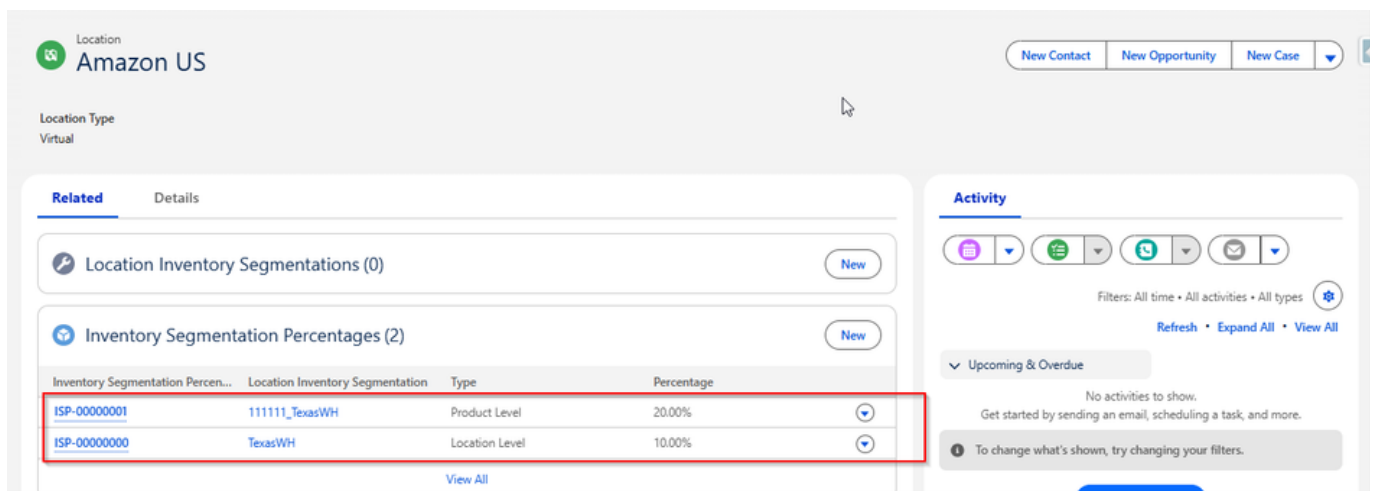
Owner: Johnson Nadar

* Location: (Location)

* Percentage: (Percentage)

Buttons: Cancel, Save & New, Save

Location Inventory Segmentation	The segmentation Type created in the physical Location
Location	The Virtual/Logical Location
Percentage	Inventory segmentation percentage.



Location: Amazon US

Location Type: Virtual

Related | Details

Location Inventory Segmentations (0) [New](#)

Inventory Segmentation Percentages (2) [New](#)

Inventory Segmentation Percen...	Location Inventory Segmentation	Type	Percentage
ISP-00000001	111111_TexasWH	Product Level	20.00%
ISP-00000000	TexasWH	Location Level	10.00%

[View All](#)

Activity

Filters: All time • All activities • All types [Settings](#)

[Refresh](#) • [Expand All](#) • [View All](#)

Upcoming & Overdue

No activities to show.
Get started by sending an email, scheduling a task, and more.

To change what's shown, try changing your filters.

Upload Inventory Service

Endpoint

```
/services/apexrest/perfawarellc/upload-inventory
```

Input

```
{
  "externalIdentifier": "2026012102",
  "records": [
    {
      "recordId": "0a87539d-1111-47bc-444-11152e39e0003",
      "sku": "111112",
      "locationId": "TexasWH",
      "effectiveDate": "2025-12-23T00:00:00.790896-07:00",
      "onHand": 100,
      "safetyStockCount": 0
    }
  ]
}
```

Output

```
{
  "uploadId": "5d6091b3-c05b-4ab4-90f4-3e577ca3e882",
  "uploadHistoryId": "a04fj00000018S1IAAE",
  "success": true,
  "responseCode": 201,
  "message": "Inventory uploaded successfully.",
  "description": ""
}
```

Response Codes

Code	Description
201	Accepted
500	Internal Error
206	Partially Accepted

Permission Sets

- 1.DOM Routing Permission Set (DOM Routing related permissions)
- 2.Inventory Segmentation Permission Set (Inventory segmentation related permissions)