

SF-LWR Extension Framework : Flat Shipping Rates

[Functional Overview](#)

[Technical Details](#)

[Sequence Diagram](#)

[Configuration details](#)

[Record - Internal Shipping Provider\(Shipping Provider \)](#)

[Custom Metadata Type - Internal Shipping Rates](#)

[Record \(example\)- Shipping 2-3 Days \(Internal Shipping Rates \)](#)

[Source Code](#)

Functional Overview

This document provides details about configuring flat/internal rates and combining them with other external rates (if required).

For common details you can refer to parent doc [here](#).

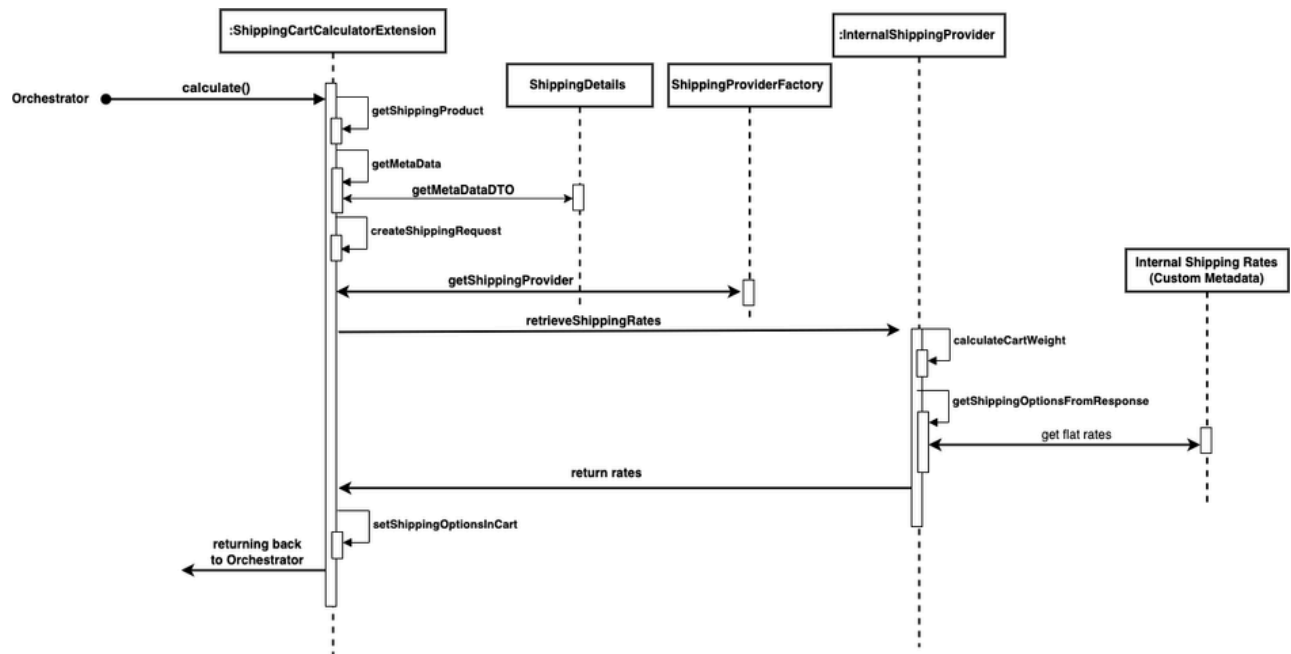
Technical Details

This integration implementation has two parts :

- **Configuration**
 - Custom Meta Data Records
 - Internal Shipping Provider
 - Custom Metadata Type (used to store flat rates)
 - Internal Shipping Rates
- **Source Code**
 - InternalShippingProvider

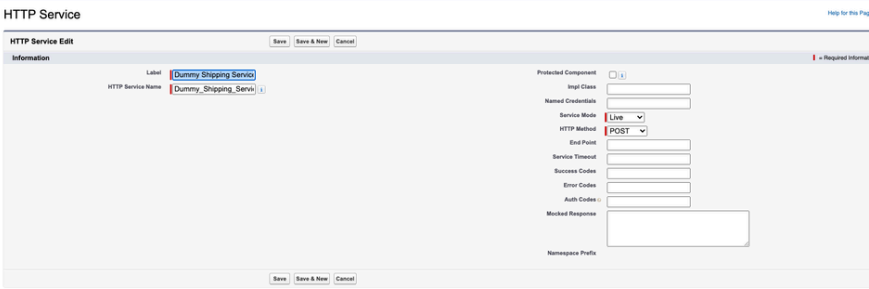
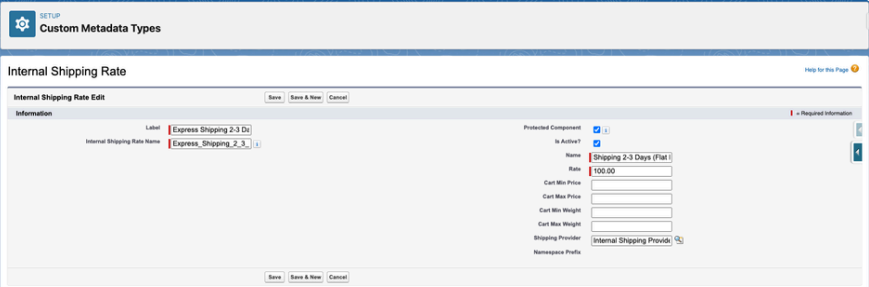
*Note : For configuring internal/flat rates, have created a new metadata type “**Internal Shipping Rates**”*

Sequence Diagram



Configuration details

| | Section | Task | Details |
|---|----------------|--|---|
| 1 | Salesforce Org | Shipping Provider - Create a record for Flat shipping rates | <p>Create a record Internal Shipping Provider for Custom Meta Data Type Shipping Provider</p> |
| 2 | Salesforce Org | HTTP Service - Create a record for Flat shipping rates | <p>Create a record - Dummy Shipping Service of Custom Meta Data Type - HTTP Service</p> <p>Note : As in actual there will not be any callout , just creating a dummy service.</p> |

| | | | |
|---|----------------|--|---|
| | | |  <p>HTTP Service Edit</p> <p>Information</p> <p>Label: Dummy Shipping Service</p> <p>HTTP Service Name: Dummy_Shipping_Servi</p> <p>Protected Component: <input type="checkbox"/></p> <p>Impl Class: <input type="text"/></p> <p>Named Credentials: <input type="text"/></p> <p>Service Mode: Live</p> <p>HTTP Method: POST</p> <p>URL Prefix: <input type="text"/></p> <p>Service Timeout: <input type="text"/></p> <p>Success Codes: <input type="text"/></p> <p>Error Codes: <input type="text"/></p> <p>Auth Codes: <input type="text"/></p> <p>Mocked Responses: <input type="text"/></p> <p>Namespace Prefix: <input type="text"/></p> <p>Buttons: Save, Save & New, Cancel</p> |
| 3 | Salesforce Org | Internal Shipping Rates - Create a record for Flat shipping rates | <p>Create a record for Custom Meta Data Type Internal Shipping Rates</p> <p>Note : You can create multiple rates as per requirement</p>  <p>Internal Shipping Rate Edit</p> <p>Information</p> <p>Label: Express Shipping 2-3 Days</p> <p>Internal Shipping Rate Name: Express_Shipping_2_3</p> <p>Protected Component: <input checked="" type="checkbox"/></p> <p>Is Active? <input checked="" type="checkbox"/></p> <p>Name: Shipping 2-3 Days (Flat)</p> <p>Rate: 100.00</p> <p>Cart Min Price: <input type="text"/></p> <p>Cart Max Price: <input type="text"/></p> <p>Cart Min Weight: <input type="text"/></p> <p>Cart Max Weight: <input type="text"/></p> <p>Shipping Provider: Internal Shipping Provider</p> <p>Namespace Prefix: <input type="text"/></p> <p>Buttons: Save, Save & New, Cancel</p> |

Record - Internal Shipping Provider(Shipping Provider)

| Field Label | Field Value |
|----------------------|---------------------------------|
| Is Active? | true |
| Context Name | <Webstore name> e.g. R1Training |
| Context Locale | <locale> e.g. en_US |
| Impl Class | InternalShippingProvider |
| Max Weight Threshold | N/A |
| Min Weight Threshold | N/A |
| Service | Dummy Shipping Service |
| Account Number | N/A |
| Shipping Options | N/A |
| Request | N/A |

Custom Metadata Type - Internal Shipping Rates

| Field Label | API Name | Data Type | Notes |
|-------------|----------|-----------|-------|
|-------------|----------|-----------|-------|

| | | | |
|-------------------|----------------------|--|---|
| Cart Max Price | Cart_Max_Price__c | Number(16, 2) | To define a price amount range for which a particular shipping method will be used , this field stores Max cart/order amount value |
| Cart Max Weight | Cart_Max_Weight__c | Number(16, 2) | To define a weight range for which a particular shipping method will be used , this field stores Max cart/order weight |
| Cart Min Price | Cart_Min_Price__c | Number(16, 2) | To define a price amount range for which a particular shipping method will be used , this field stores Min cart/order amount value |
| Cart Min Weight | Cart_Min_Weight__c | Number(16, 2) | To define a weight range for which a particular shipping method will be used , this field stores Min cart/order weight |
| Is Active? | Is_Active__c | Checkbox | Make sure to activate a shipping method to be used |
| Name | Name__c | Text(255) (Unique Case Insensitive) | Name of a shipping method/option (it will be displayed on UI) |
| Rate | Rate__c | Number(16, 2) | Cost of a internal shipping method/option |
| Shipping Provider | Shipping_Provider__c | Metadata Relationship(Shipping Provider) | This associates rate record with corresponding Shipping Provider record |

Record (example)- Shipping 2-3 Days (Internal Shipping Rates)

| Field Label | Field Value |
|-------------------|----------------------------|
| Name | Shipping 2-3 Days |
| Rate | 100 |
| Is Active? | true |
| Shipping Provider | Internal Shipping Provider |
| Cart Max Price | |
| Cart Max Weight | |
| Cart Min Price | |
| Cart Min Weight | |

Source Code

| Class | Details | Code |
|--------------------------|--|--|
| InternalShippingProvider | This class extends ShippingProvider class & implements following methods - | <pre> 1 public class InternalShippingProvider extends ShippingProvider 2 public InternalShippingProvider() { 3 } 4 public override Map<String, ShippingProviderResponse> retrieve 5 ShippingProviderRequest shippingRequest </pre> |

prepareRequestBody

getShippingOptionsFromResponse

```
6    ) {
7        List<String> responseList = new List<String>();
8        Map<String, ShippingProviderResponse> shippingMethodsWithRate = new HashMap<>();
9        Decimal totalCartWeight = 0;
10       try {
11           List<Decimal> lstShippingWeight = calculateCartWeight(shippingRequest);
12           for (Decimal weight : lstShippingWeight) {
13               totalCartWeight += weight;
14           }
15           shippingRequest.packageWeight = totalCartWeight;
16           shippingMethodsWithRate = getShippingOptionsFromResponse(responseList, shippingRequest);
17       } catch (Exception expObj) {
18           System.debug(
19               'Exception due to error ==== ' +
20               expObj.getMessage() +
21               'at Line Number ==== ' +
22               expObj.getLineNumber()
23           );
24           // WK_Exception.log(expObj, applicationName, moduleName,
25           // methodName, supportData);
26       }
27       return shippingMethodsWithRate;
28   }
29
30   public override Map<String, ShippingProviderResponse> getShippingOptionsFromResponse(
31       List<String> responseList,
32       ShippingProviderRequest shippingRequest
33   ) {
34       Map<String, ShippingProviderResponse> shippingMethodsWithRate = new HashMap<>();
35       List<Internal_Shipping_Rate__mdt> shippingRates = [
36           SELECT
37               Id,
38               Name__c,
39               Rate__c,
40               Cart_Max_Price__c,
41               Cart_Max_Weight__c,
42               Cart_Min_Price__c,
43               Cart_Min_Weight__c,
44               Shipping_Provider__c
45           FROM Internal_Shipping_Rate__mdt
46           WHERE
47               Shipping_Provider__c = :shippingRequest.shippingMetaDatum
48               AND Is_Active__c = TRUE
49       ];
50       ShippingProviderResponse shippingResponse;
51       if (shippingRates != null && shippingRates.size() > 0) {
52           for (Internal_Shipping_Rate__mdt rate : shippingRates) {
53               shippingResponse = new ShippingProviderResponse();
54               shippingResponse.serviceCode = rate.Name__c;
55               shippingResponse.cost = rate.Rate__c;
56               shippingResponse.carrier = shippingRequest.shippingMetaDatum;
57               shippingResponse.displayOrder = shippingRequest.shippingMetaDatum;
58               shippingMethodsWithRate.put(
59                   shippingResponse.serviceCode,
60                   shippingResponse
61               );
62           }
63       }
64   }
```

```
66     }
67     return shippingMethodsWithRate;
68 }
69 public override void prepareRequestBody(
70     ShippingProviderRequest shippingRequest,
71     Map<String, String> callOutRequest
72 ) {
73 }
74 }
75
```