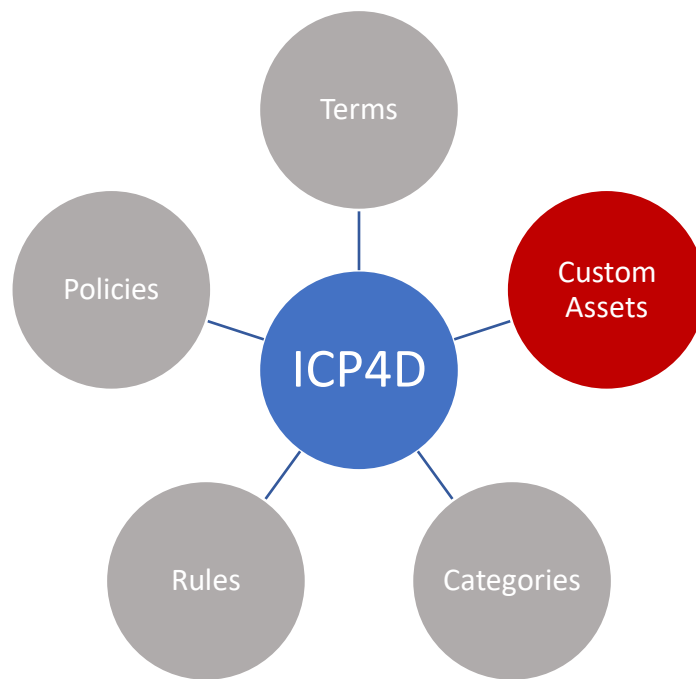


## Using The Custom Asset Bundle Utility

ICP4D allows the users to manage information governance for various assets like policies, terms, rules, and categories out-of-the-box. In addition to these assets, ICP4D also allows the user to upload custom assets.



Before uploading the instances of these Custom Assets, we would need to perform the following steps:

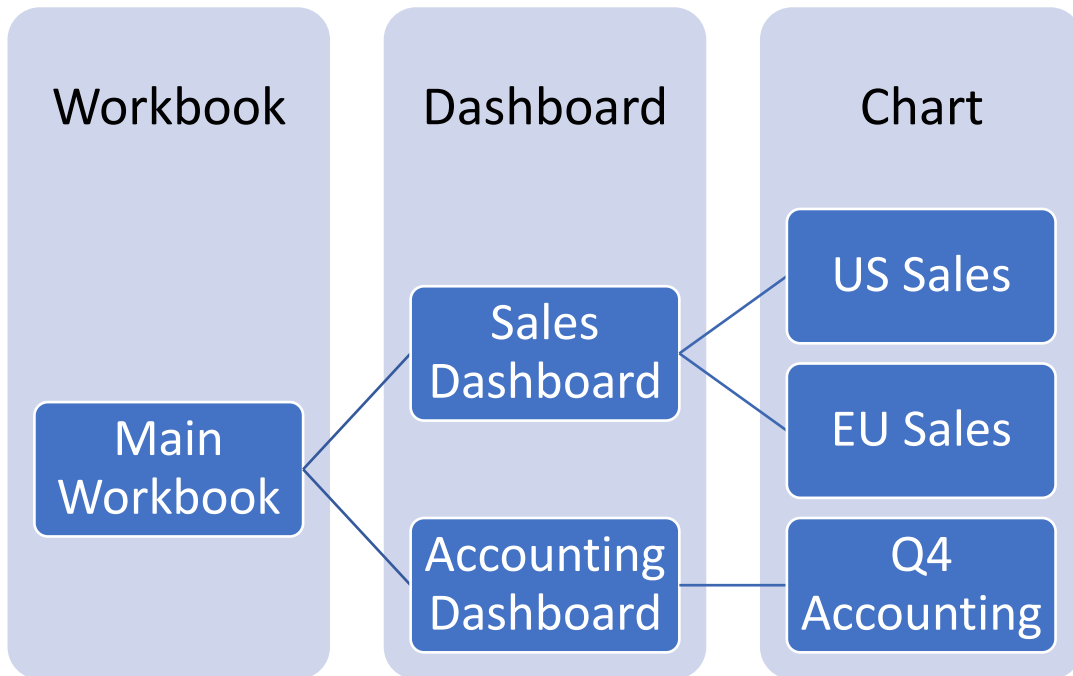
1. Identify the type of assets that we want to upload, the attributes that these assets may have, and any other type assets that it may contain or be a part of.
2. Create an asset bundle with the information obtained from the step above and upload it to ICP-4D. This process of uploading the asset bundle is known as '*registering*' the asset bundle.

This article focuses on the steps for registering the asset bundle and using the Custom Bundle utility to create the asset bundle.

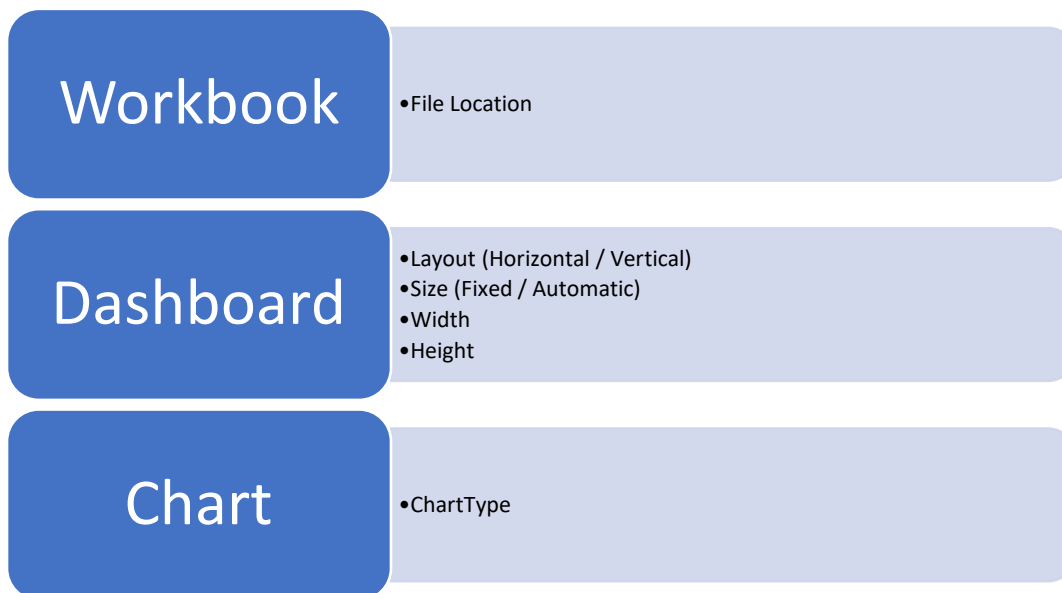
Example of a Custom Asset Bundle :

Let us consider a fictitious BI tool '**DVPro**', we wish to map the structure of the assets for this tool and upload it into ICP-4D.

This tool may have individual files called as Workbooks, further each workbook may have multiple dashboards and each dashboard may be composed of various views. An example of this is given below.



We wish to define the structure for this asset bundle. The assets belonging to this asset bundle and their various attributes can be described as mentioned below

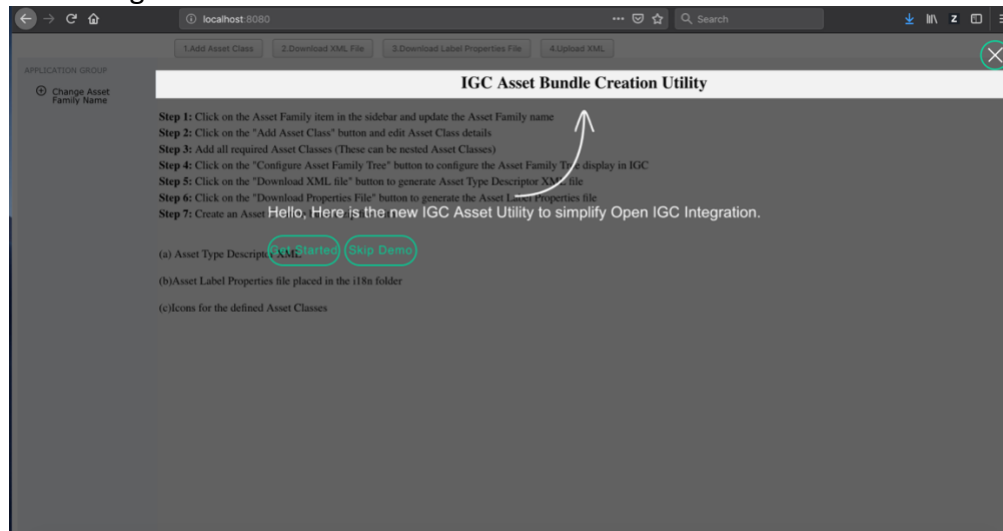


Let us now have a look at how this Custom Asset Bundle can be created using the Custom Asset Bundle Utility.

## 1 : Installing the Custom Bundle Utility

Prerequisites: This Utility requires [Docker](#) to be installed on the host machine.

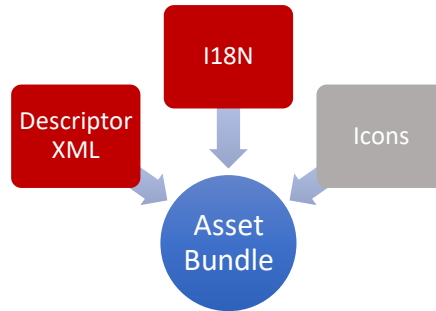
- i. Pull the latest docker image using the command  
**docker pull ibmicp4d/custom-bundle-utility**
- ii. Run the utility using the command  
**docker run -p 8080:8080 --rm -it ibmicp4d/custom-bundle-utility**
- iii. Now, open a browser window and navigate to **localhost:8080**. You should see a screen like the figure below.



## 2 : Structure of Asset Bundle and creating the XML File

The asset bundle is uploaded as a zip file into ICP4D. It consists of the following files and folders

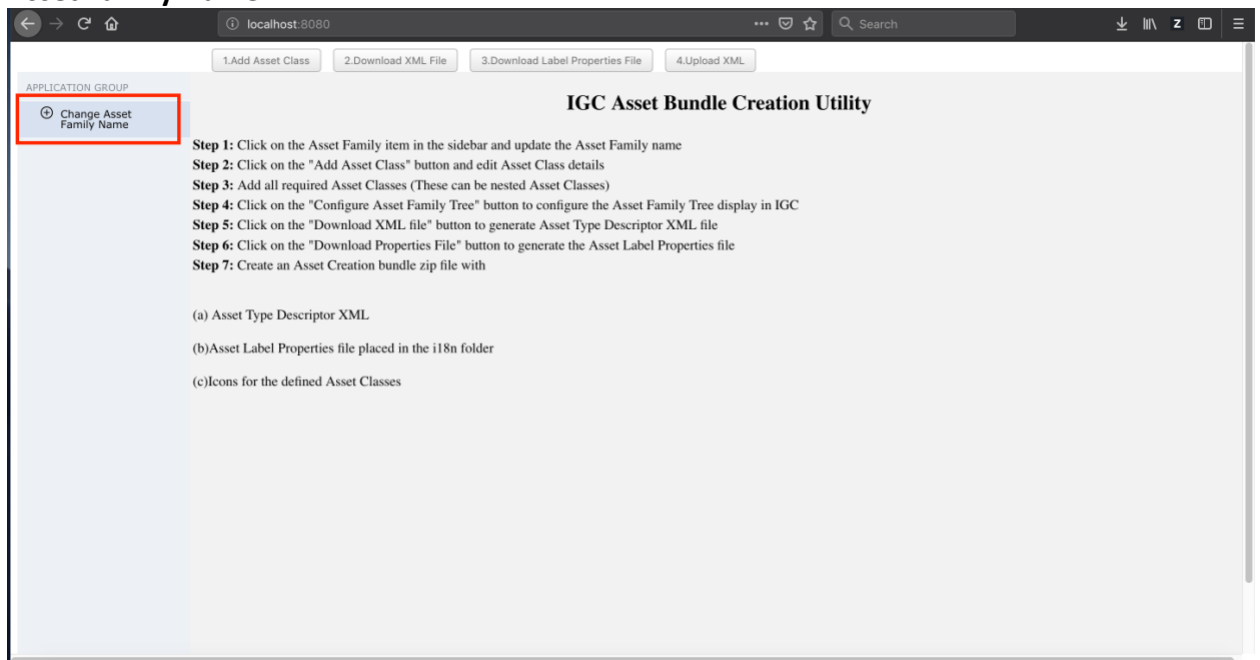
1. asset\_type\_descriptor.xml : This file describes the entire structure of the Custom Asset Bundle
2. i18n : This folder contains the files which are required to translate the asset labels into different languages
3. icons : This folder contains icons for the various asset types.



The utility helps in the creation of the asset\_type\_descriptor and the i18n file for the labels.

Let us now create this bundle using the utility.

1. Navigate to **localhost:8080/**. Click on the **X** checkmark to skip the demo. Click on **Change Asset Family Name**.



2. Enter the name of the Bundle and click on the Save Button

1.Add Asset Class 2.Download XML File 3.Download Label Properties File 4.Upload XML

APPLICATION GROUP

DVPro

Asset Family Details

Application Details

Open IGC Bundle N... DVPro

Display Position 1

Contained Assets Comma separated asset type list

Assets Label Key ASSET\_FMLY\_LBL\_KEY

Assets Label DVPro

Save

XML Content <?xml version='1.0' encoding='UTF-8' standalone='yes'?><descriptor xmlns='http://www.ibm.com/ils/igc/asset-type-descriptor' bundleId='DVPro'>

- Click on 'Add Asset Class' button and enter the name of the first Asset class (In our case it would be 'Workbook'). In addition to that, also enter the label and the plural label for the asset class. The label and the plural label key values are used if the asset name need to be translated into different languages.

1.Add Asset Class 2.Download XML File 3.Download Label Properties File 4.Upload XML

APPLICATION GROUP

DVPro

Asset Class Details

General Attribute 1 Attribute 2 Attribute 3 Attribute 4 Attribute 5 Attribute 6 Attribute 7 Attribute 8 Attribute 9 Attribute 10

ID AssetClass\_1

Name Workbook

Short Description

Long Description

Super Class Ref

Asset Container Class Ref

Operational Meta data Class Ref

Can Have Image ☒

Expandable In Lineage Graphs ☒

Data Access Role Active

Display Position 1

Label Key ASSET\_CLASS1\_LBL\_KEY

Label Workbook

Plural Label Key ASSET\_CLASS1\_LBL\_S\_KEY

Plural Label Workbooks

Save

XML Content <?xml version='1.0' encoding='UTF-8' standalone='yes'?><descriptor xmlns='http://www.ibm.com/ils/igc/asset-type-descriptor' bundleId='DVPro'>

- Click on the 'Attribute 1' tab and enter the name of the attribute (Location)

1.Add Asset Class 2.Download XML File 3.Download Label Properties File 4.Upload XML

APPLICATION GROUP  
- DVPro  
→ Workbook

Asset Class Details

General Attribute 1 Attribute 2 Attribute 3 Attribute 4 Attribute 5 Attribute 6 Attribute 7 Attribute 8 Attribute 9 Attribute 10

ID ASSET\_CLASS1\_ATTR1\_ID Type String  
Multivalued ☐ Editable ☒  
Label Key ASSET\_CLASS1\_ATTR1\_LBL\_KEY Label Location  
ID ASSET\_CLASS1\_ATTR1\_VAL1\_ID ID ASSET\_CLASS1\_ATTR1\_VAL2\_ID  
Key ASSET\_CLASS1\_ATTR1\_VAL1\_LBL\_KEY Key ASSET\_CLASS1\_ATTR1\_VAL2\_LBL\_KEY  
Label Label  
ID ASSET\_CLASS1\_ATTR1\_VAL3\_ID ID ASSET\_CLASS1\_ATTR1\_VAL4\_ID  
Key ASSET\_CLASS1\_ATTR1\_VAL3\_LBL\_KEY Key ASSET\_CLASS1\_ATTR1\_VAL4\_LBL\_KEY  
Label Label  
ID ASSET\_CLASS1\_ATTR1\_VAL5\_ID  
Key ASSET\_CLASS1\_ATTR1\_VAL5\_LBL\_KEY  
Label

Save

XML Content <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<descriptor xmlns="http://www.ibm.com/its/igc/asset-type-descriptor" bundleId="DVPro">

- Click on the Save button. Note that the asset\_type\_descriptor.xml file is getting generated in the XML Content section at the bottom of the page. At any point, you can save this partially generated XML using the 'Download XML button' and resume its creation later using the 'Upload XML' button

1.Add Asset Class 2.Download XML File 3.Download Label Properties File 4.Upload XML

APPLICATION GROUP  
- DVPro  
→ Workbook

Asset Class Details

General Attribute 1 Attribute 2 Attribute 3 Attribute 4 Attribute 5 Attribute 6 Attribute 7 Attribute 8 Attribute 9 Attribute 10

ID ASSET\_CLASS1\_ATTR1\_ID Type String  
Multivalued ☐ Editable ☒  
Label Key ASSET\_CLASS1\_ATTR1\_LBL\_KEY Label Location  
ID ASSET\_CLASS1\_ATTR1\_VAL1\_ID ID ASSET\_CLASS1\_ATTR1\_VAL2\_ID  
Key ASSET\_CLASS1\_ATTR1\_VAL1\_LBL\_KEY Key ASSET\_CLASS1\_ATTR1\_VAL2\_LBL\_KEY  
Label Label  
ID ASSET\_CLASS1\_ATTR1\_VAL3\_ID ID ASSET\_CLASS1\_ATTR1\_VAL4\_ID  
Key ASSET\_CLASS1\_ATTR1\_VAL3\_LBL\_KEY Key ASSET\_CLASS1\_ATTR1\_VAL4\_LBL\_KEY  
Label Label  
ID ASSET\_CLASS1\_ATTR1\_VAL5\_ID  
Key ASSET\_CLASS1\_ATTR1\_VAL5\_LBL\_KEY  
Label

Save

XML Content <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<descriptor xmlns="http://www.ibm.com/its/igc/asset-type-descriptor" bundleId="DVPro">  
<family position="1" classRef="Comma separated asset type list">  
<label key="ASSET\_FWK1\_LBL\_KEY" isFunctional="true"/>  
</family>

- Once we have the structure of the first asset defined, click on the 'Add Asset Class' button again to describe the next asset ('Dashboard')



APPLICATION GROUP

- DVPro
  - Workbook
  - Dashboard

Asset Class Details

General Attribute 1 Attribute 2 Attribute 3 Attribute 4 Attribute 5 Attribute 6 Attribute 7 Attribute 8 Attribute 9 Attribute 10

ID: ASSET\_CLASS2\_ATTR2\_ID Type: String

Multivalued: ☐ Editable: ☒

Label Key: ASSET\_CLASS2\_ATTR2\_LBL\_KEY Label: Size

ID: ASSET\_CLASS2\_ATTR2\_VAL1\_ID ID: ASSET\_CLASS2\_ATTR2\_VAL2\_ID

Key: ASSET\_CLASS2\_ATTR2\_VAL1\_LBL\_KEY Key: ASSET\_CLASS2\_ATTR2\_VAL2\_LBL\_KEY

Label: Fixed Label: Automatic

ID: ID

Key: Key

Label: Label

ID: ID

Key: Key

Label: Label

Save

XML Content: <?xml version="1.0" encoding="UTF-8" standalone="yes"?><descriptor xmlns="http://www.ibm.com/ila/lqc/asset-type-descriptor" handleId="DVPro"><family position="1" classRef="AssetClass\_1" >

- If the dashboard has a fixed size, you may want to specify the Width and the Height of the dashboard. Click on the 'Attribute 3' tab to add the Width attribute. This attribute would have a numeric value, so select the 'Type' drop down and click on 'Double'

APPLICATION GROUP

- DVPro
  - Workbook
  - Dashboard
  - Chart

Asset Class Details

General Attribute 1 Attribute 2 Attribute 3 Attribute 4 Attribute 5 Attribute 6 Attribute 7 Attribute 8 Attribute 9 Attribute 10

ID: ASSET\_CLASS2\_ATTR3\_ID Type: Double

Multivalued: ☐ Editable: ☐

Label Key: ASSET\_CLASS2\_ATTR3\_LBL\_KEY Label: Double

ID: ASSET\_CLASS2\_ATTR3\_VAL1\_ID ID: ASSET\_CLASS2\_ATTR3\_VAL2\_ID

Key: ASSET\_CLASS2\_ATTR3\_VAL1\_LBL\_KEY Key: ASSET\_CLASS2\_ATTR3\_VAL2\_LBL\_KEY

Label: Label Label: Label

ID: ASSET\_CLASS2\_ATTR3\_VAL3\_ID ID: ASSET\_CLASS2\_ATTR3\_VAL4\_ID

Key: ASSET\_CLASS2\_ATTR3\_VAL3\_LBL\_KEY Key: ASSET\_CLASS2\_ATTR3\_VAL4\_LBL\_KEY

Label: Label Label: Label

ID: ASSET\_CLASS2\_ATTR3\_VAL5\_ID

Key: ASSET\_CLASS2\_ATTR3\_VAL5\_LBL\_KEY

Label: Label

Save

XML Content: <?xml version="1.0" encoding="UTF-8" standalone="yes"?><descriptor xmlns="http://www.ibm.com/ila/lqc/asset-type-descriptor" handleId="DVPro"><family position="1" classRef="AssetClass\_2.AssetClass\_1.AssetClass\_3" >

Also, specify the name of the attribute ('Width')



The screenshot shows the 'Asset Class Details' form for 'Attribute 3'. The 'Label' field is highlighted with a red box and contains the text 'Width'. The 'Type' is set to 'Double'. The 'Editable' checkbox is checked. The 'ID' field contains 'ASSET\_CLASS2\_ATTR3\_ID'. The 'Label Key' field contains 'ASSET\_CLASS2\_ATTR3\_LBL\_KEY'. The 'Multivalued' checkbox is unchecked. The 'XML Content' field at the bottom shows the following XML snippet:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<descriptor xmlns="http://www.ibm.com/its/lqc/asset-type-descriptor" handleId="DVPro">
<family position="1" classRefs="AssetClass_2,AssetClass_1,AssetClass_3">
```

10. Click on Attribute 4, and repeat the step above for 'Height'

The screenshot shows the 'Asset Class Details' form for 'Attribute 4'. The 'Label' field is highlighted with a red box and contains the text 'Height'. The 'Type' is set to 'Float'. The 'Editable' checkbox is checked. The 'ID' field contains 'ASSET\_CLASS2\_ATTR4\_ID'. The 'Label Key' field contains 'ASSET\_CLASS2\_ATTR4\_LBL\_KEY'. The 'Multivalued' checkbox is unchecked. The 'XML Content' field at the bottom shows the following XML snippet:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<descriptor xmlns="http://www.ibm.com/its/lqc/asset-type-descriptor" handleId="DVPro">
<family position="1" classRefs="AssetClass_1">
```

11. Click on the 'Save' button when all the asset attributes are defined.

12. Click on 'Add Asset Class' button to add the final asset type 'Chart', click on the 'General' tab and specify the name, and the labels.

1.Add Asset Class 2.Download XML File 3.Download Label Properties File 4.Upload XML

APPLICATION GROUP

- DVPro
  - Workbook
  - Dashboard
    - Chart

Asset Class Details

General Attribute 1 Attribute 2 Attribute 3 Attribute 4 Attribute 5 Attribute 6 Attribute 7 Attribute 8 Attribute 9 Attribute 10

ID: AssetClass\_3 Data Access Role: Active

Name: Chart Display Position: 1

Short Description: Label Key: ASSET\_CLASS3\_LBL\_KEY

Long Description: Label: Chart

Super Class Ref: Plural Label Key: ASSET\_CLASS3\_LBL5\_KEY

Asset Container Class Ref: AssetClass\_2 Plural Label: Charts

Operational Meta data Class Ref:

Can Have Image: ☒

Expandable In Lineage Graphs: ☒

Save

XML Content: <?xml version="1.0" encoding="UTF-8" standalone="yes"?><descriptor xmlns="http://www.ibm.com/ila/lqc/asset-type-descriptor" bundleId="DVPro"><family position="1" classRef="AssetClass\_2,AssetClass\_1">

13. Click on 'Attribute 1' tab to add the ChartType Attribute.

1.Add Asset Class 2.Download XML File 3.Download Label Properties File 4.Upload XML

APPLICATION GROUP

- DVPro
  - Workbook
  - Dashboard
    - Chart

Asset Class Details

General Attribute 1 Attribute 2 Attribute 3 Attribute 4 Attribute 5 Attribute 6 Attribute 7 Attribute 8 Attribute 9 Attribute 10

ID: ASSET\_CLASS3\_ATTR1\_ID Type: String

Multivalued: ☐ Editable: ☒

Label Key: ASSET\_CLASS3\_ATTR1\_LBL\_KEY Label: ChartType

ID: ASSET\_CLASS3\_ATTR1\_VAL1\_ID ID: ASSET\_CLASS3\_ATTR1\_VAL2\_ID

Key: ASSET\_CLASS3\_ATTR1\_VAL1\_LBL\_KEY Key: ASSET\_CLASS3\_ATTR1\_VAL2\_LBL\_KEY

Label: Label

ID: ASSET\_CLASS3\_ATTR1\_VAL3\_ID ID: ASSET\_CLASS3\_ATTR1\_VAL4\_ID

Key: ASSET\_CLASS3\_ATTR1\_VAL3\_LBL\_KEY Key: ASSET\_CLASS3\_ATTR1\_VAL4\_LBL\_KEY

Label: Label

ID: ASSET\_CLASS3\_ATTR1\_VAL5\_ID

Key: ASSET\_CLASS3\_ATTR1\_VAL5\_LBL\_KEY

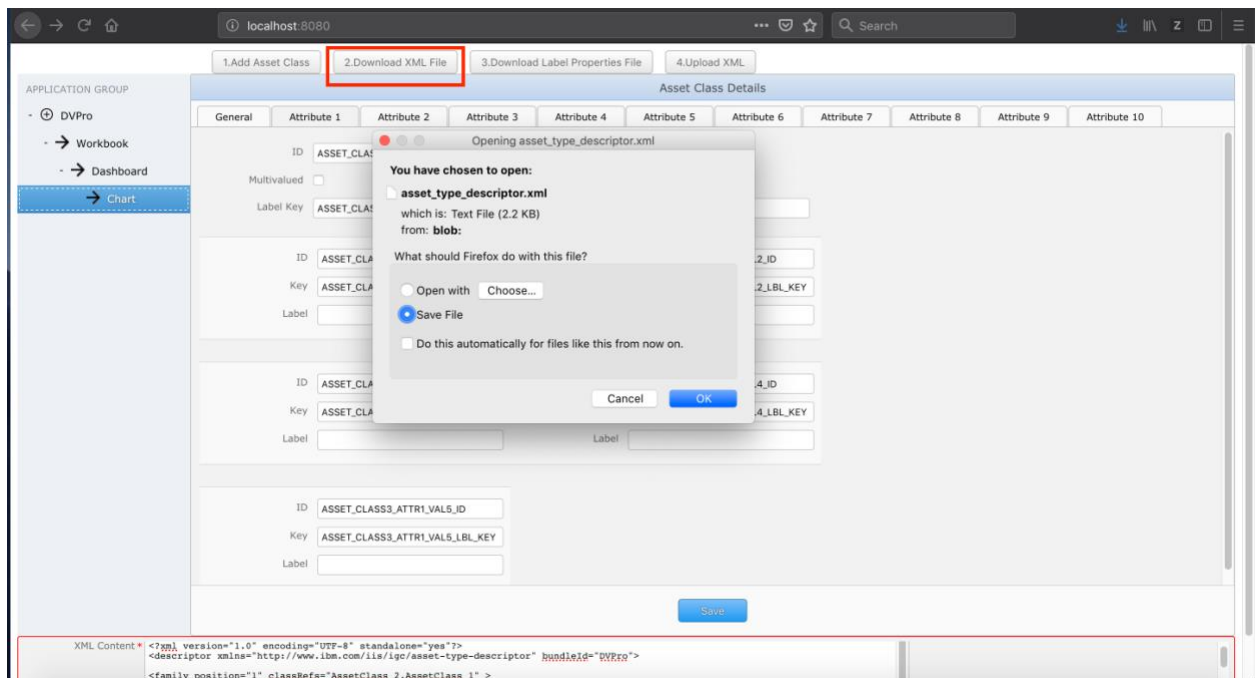
Label: Label

Save

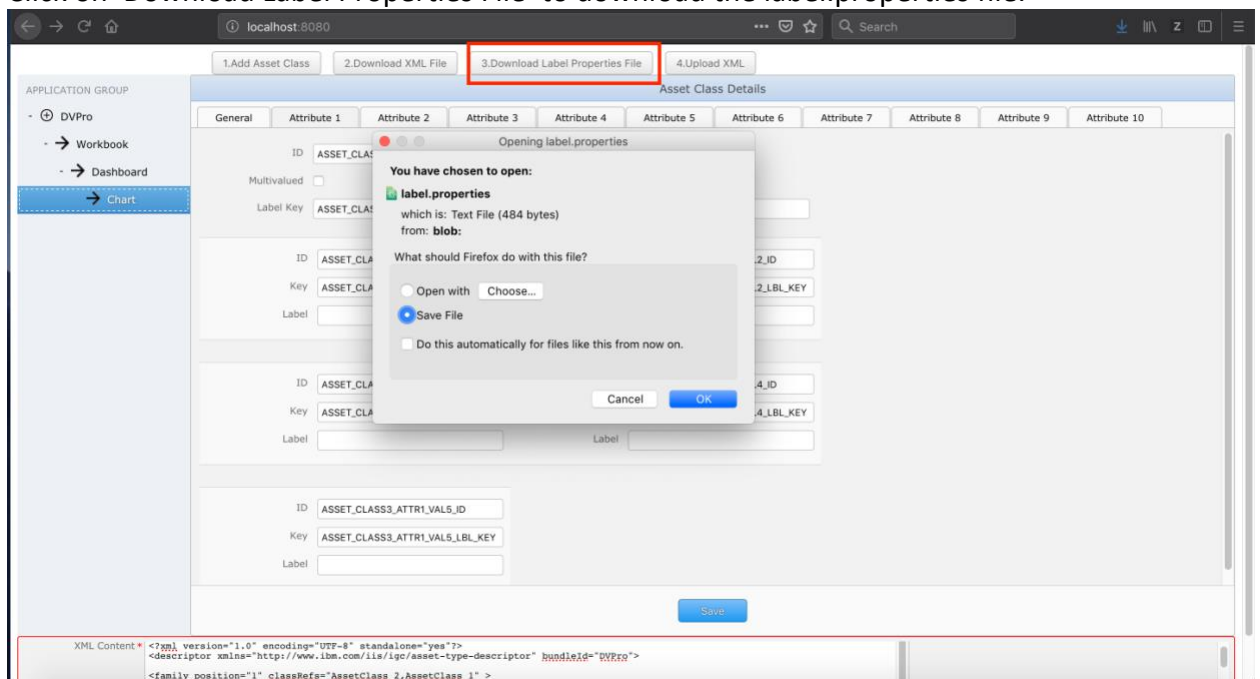
XML Content: <?xml version="1.0" encoding="UTF-8" standalone="yes"?><descriptor xmlns="http://www.ibm.com/ila/lqc/asset-type-descriptor" bundleId="DVPro"><family position="1" classRef="AssetClass\_2,AssetClass\_1">

14. Click on the 'Save' button

15. Once all the assets and their attributes have been defined, click on 'Download XML' button to download the 'asset\_type\_descriptor.xml' generated.

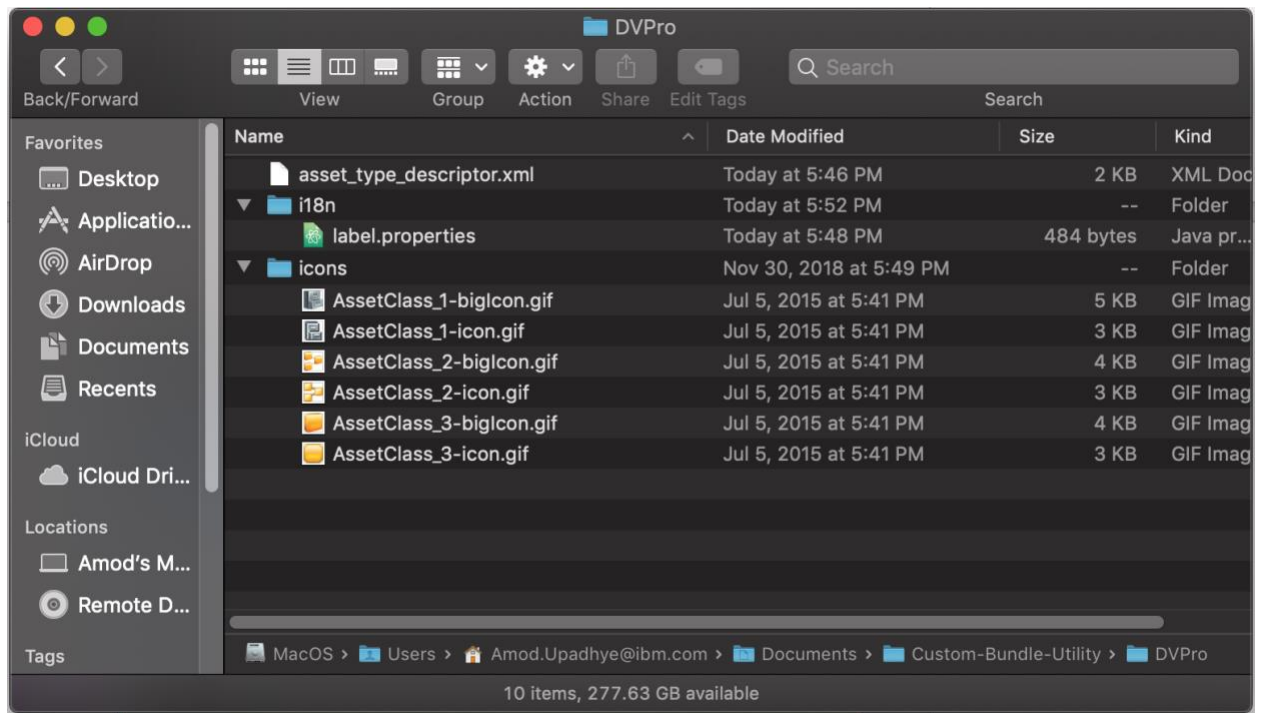


16. Click on 'Download Label Properties File' to download the label.properties file.



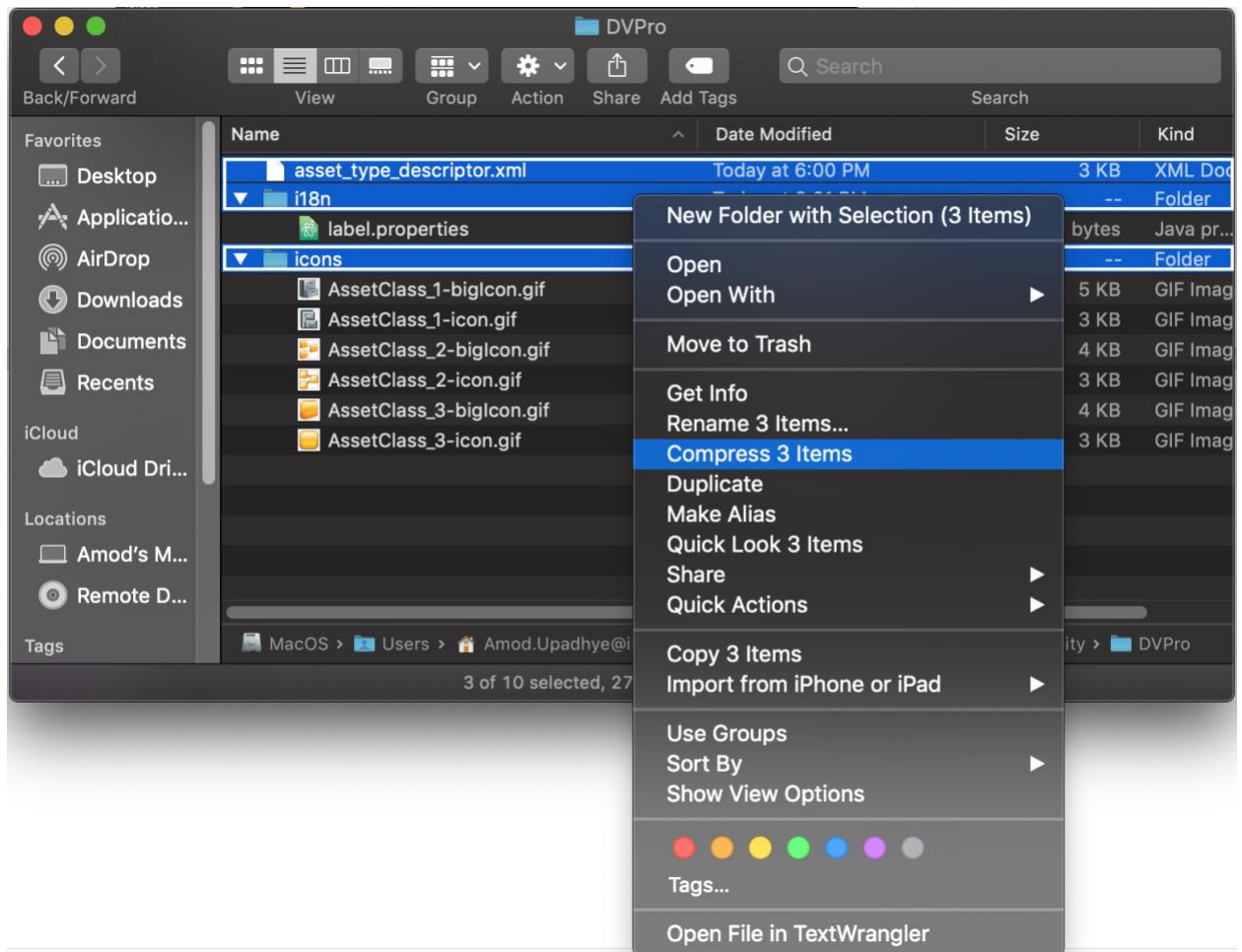
17. Once the files have been downloaded, create a folder with the asset\_type\_descriptor.xml file. Create a new folder called 'i18n' inside this folder and add the label.properties file. Create another folder called as icons which contains the graphics for the asset icons. The icons for our current example can be downloaded from [here](#).

The folder structure would look like this :



The final folder structure has also been uploaded [here](#).

18. Compress the `asset_type_descriptor.xml` file, the `i18n` folder and the `icons` folder together to create an archive bundle. The sample archive bundle can be found [here](#).



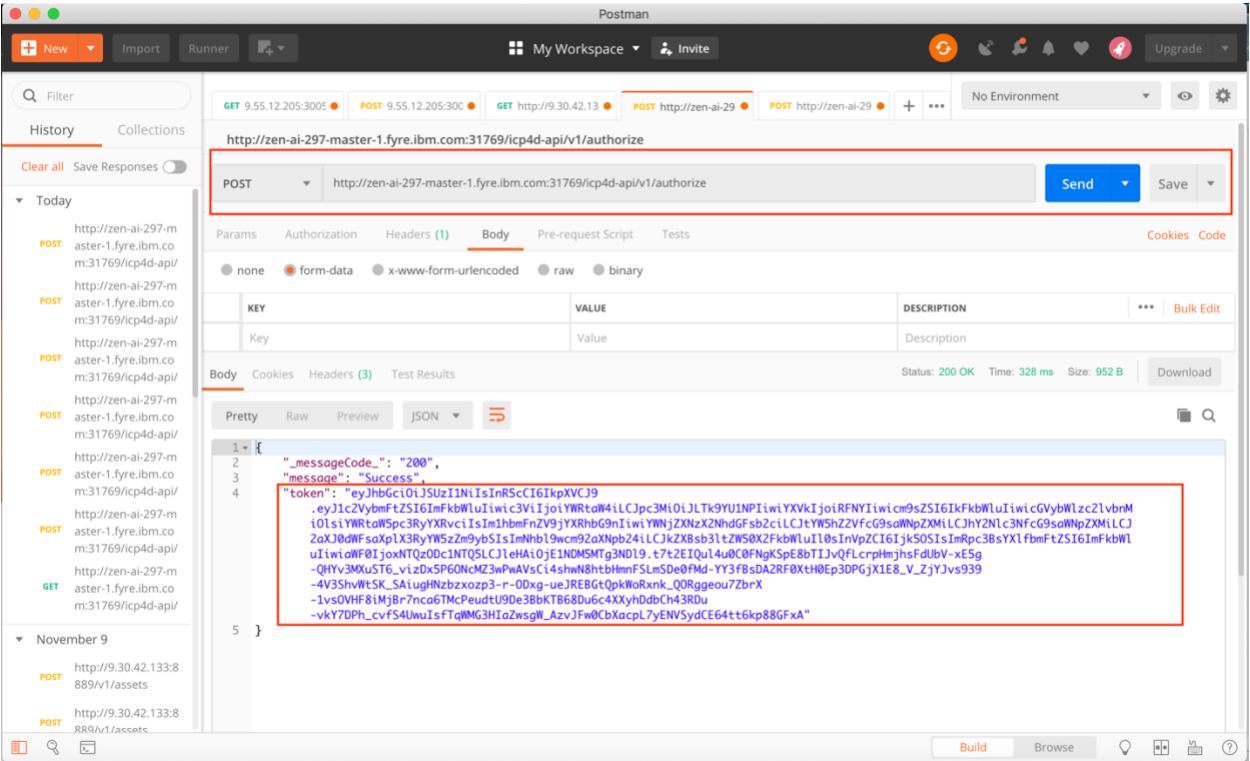
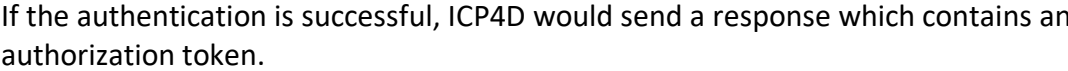
### 3: Uploading the Asset Bundle to ICP4D

In this example, we have used [Postman](#) to make the API requests to ICP4D.

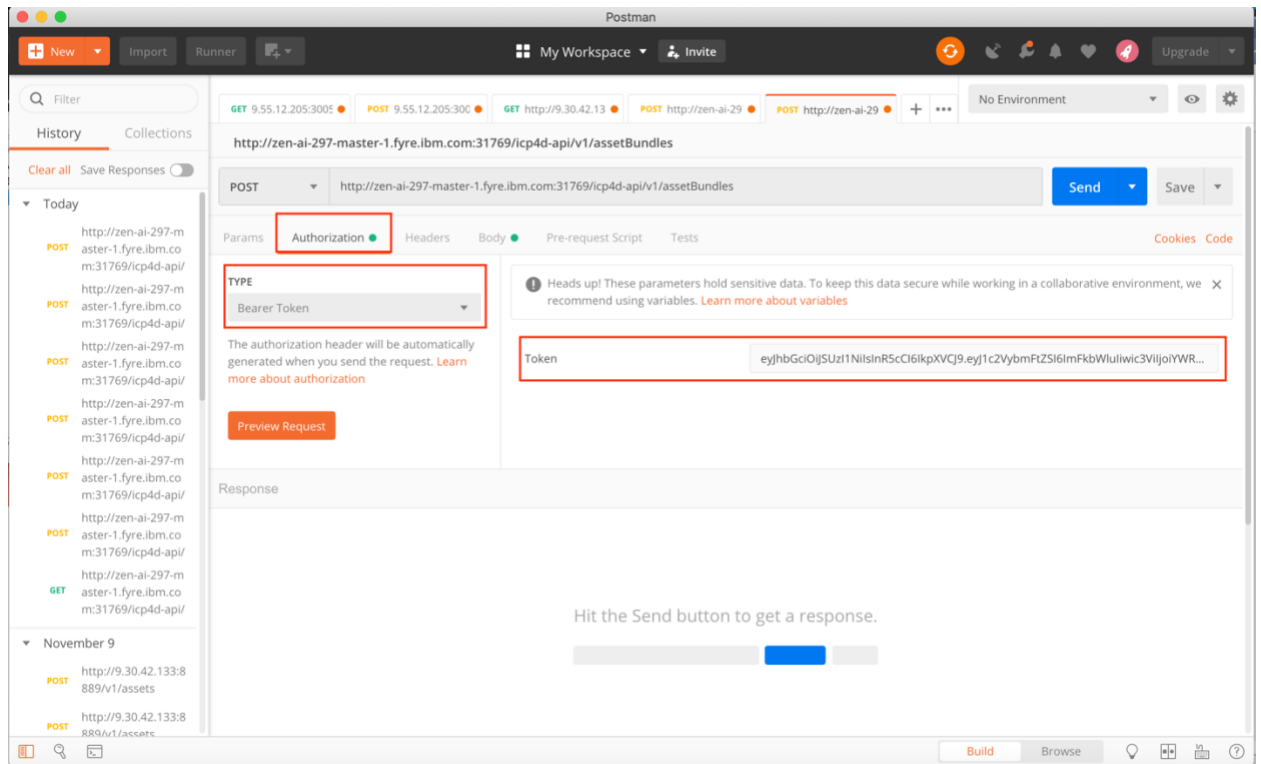
To upload the asset bundle to ICP4D, we need to do the following steps :

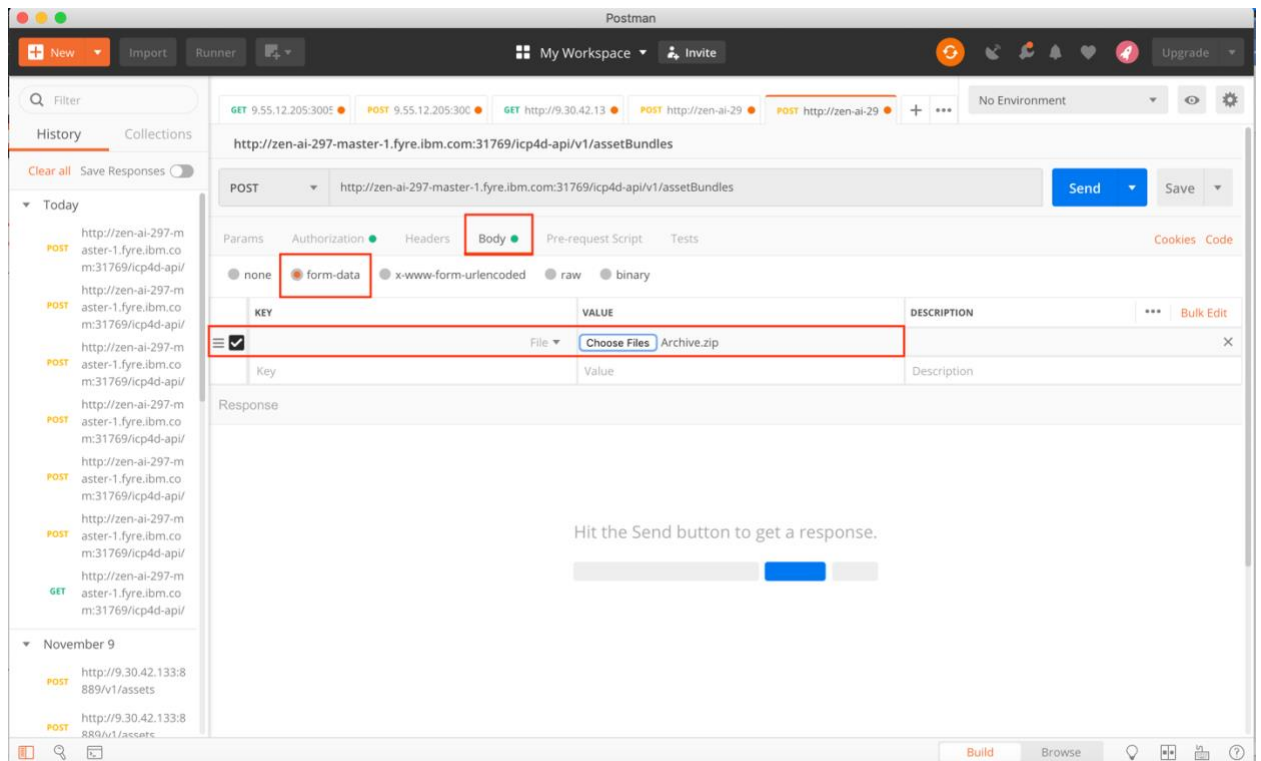
1. Obtain an Authorization token from ICP4D using '/authorize' endpoint  
For this we first need to make a POST request to '<ICPHost>/authorize' endpoint with the following json body  

```
{  
  "username": "<USERNAME>",  
  "password": "<PASSWORD>"  
}
```



2. Use the token obtained above while calling the '/assetBundles' endpoint with the assetBundle ZIP file.





With this the asset bundle is uploaded onto ICP4D and you can now begin importing custom assets belonging to this asset bundle into ICP.