Apex Share Utility

Posted on July 31, 2016 by Dilip Singh in Apex Share

We can create sharing rule in different ways like

1. Owner-Based,
2. Criteria-Based,
3. Manual Sharing and
4. Apex based sharing rule.

We will concentrate on the 4th one i.e., Apex based sharing rule as the first 3 can be done by point-N-click.

What is Apex based sharing rule?

It is similar to the first 3 types of sharing rule (as mentioned above), the only difference is we will create this sharing rule programatically and along with that we can also specify the custom reason why it is being shared.

Why Apex based sharing rule?

There are many scenarios where we all encounter business scenarios where we will opt for Apex based sharing rule (when the initial 3 are failed).

eg: Lets say I have an object “Object 1” which is being configured as “PRIVATE” in OWD.

And whenever the ownership is changed the creator lost its visibility.

Here, we will be writing a piece of apex code which shares the record with record creator.

Benefits of Apex based Sharing Rule:

1. Shared records are maintained across record owner changes
2. A record can be shared multiple times with a user or group using different Apex sharing reasons

Below is the class which can be used to implement sharing rule programatically:

```
public class SobjectApexShare {
    private static List<sObject> sObjSharelist;
private static Schema.SObjectType sObjType;
public static void shareWithCreator(String sObjShareAPIName, List<sObject> sObjReclist, Map<String,String> sObjA

        sObjSharelist = new List<sObject>();
sObjType = Schema.GetGlobalDescribe().get(sObjShareAPIName.toLowerCase());

        String sObjName = sObjShareAPIName.removeEndIgnoreCase('share');

        if(sObjName.endswith('_')){
           sObjName += 'c';
        }

        Schema.DescribeSObjectResult sObjResult = Schema.GetGlobalDescribe().get(sObjName).getDescribe();

        if(sObjResult.isCustom()){
            customObjectShare(sObjReclist,sObjAccessLevel, rowCauseReason);
        } else {
            standardObjectShare(sObjReclist,sObjAccessLevel, sObjName);
        }
        insert sObjSharelist;
    }

    private static void customObjectShare(List<sObject> sObjReclist, Map<String,String> sObjAccessLevel, String
        for(sObject sObj : sObjReclist){
             //Creating new Instance for share sObject
             sObject sObjShare = sObjType.newSObject();

             sObjShare.put('ParentId', sObj.get('id'));
             sObjShare.put('UserOrGroupId', sObj.get('CreatedById'));
             sObjShare.put('AccessLevel', sObjAccessLevel.get('AccessLevel'));
             sObjShare.put('RowCause', rowCauseReason);
             sObjSharelist.add(sObjShare);
        }
    }

    private static void standardObjectShare(String sObjName, List<sObject> sObjReclist, Map<String,String> sObjA

        for(sObject sObj : sObjReclist){
             //Creating new Instance for share sObject
             sObject sObjShare = sObjType.newSObject();

```

```
                sObjShare.put(sObjName + 'id', sObj.get('id'));
                sObjShare.put('UserOrGroupId', sObj.get('CreatedById'));

                for(String key : sObjAccessLevel.keySet()){
                    sObjShare.put(key, sObjAccessLevel.get(key));
                }
                sObjSharelist.add(sObjShare);
            }
        }
    }
```

- shareWithCreator: Main method which executes the logic to share records for both standard or custom objects programatically
    - Parameters:
        - sObjShareAPIName –> this will be the API Name of the respective share object on which we will be creating records
            - sObjReclist –> list of records which needs to be processed
            - sObjAccessLevel –> Map holding what type of access level the creator should be holding on the sharing record
            - rowCauseReason: custom sharing reason why it is being shared
    - customObjectShare: this method will be executed when the sharing record needs to be created on custom object.
    - standardObjectShare: this method will be executed when the sharing record needs to be created on standard object

Note: Currently salesforce doesn't allow to specify the custom apex share reason for standard object. In this case It will be set as Manual and the corresponding share records are deleted when owner is changed. Please vote up for enabling apex share reasons for standard object.

Here it is being shared with record creator, you can modify as per your requirement.

Psuedocode for calling shareWithCreator method:

When it is Standard Share Object

```
Map<String,String> accAccessLevel = new Map<String, String>();
accAccessLevel.put('AccountAccessLevel', 'Read');
accAccessLevel.put('OpportunityAccessLevel', 'Read');
accAccessLevel.put('CaseAccessLevel', 'Read');
system.debug('acc level values ' + accAccessLevel);

SobjectApexShare.shareWithCreator('<StandardShareObjectAPIName>', stdObjlist, accAccessLevel , null);

&nbsp;
```

// where StandardShareObjectAPIName –> eg: AccountShare

// acclist –> list of standard object records

When it is Custom Share Object

```
Map<String,String> cObjAccessLevel = new Map<String, String>();
cObjAccessLevel.put('AccessLevel', 'Edit');

SobjectApexShare.shareWithCreator('<CustomShareObjectAPIName>', custObjlist, cObjAccessLevel, Schema.Student__sha

&nbsp;
```

// custObjlist –> list of Custom object records

// CustomShareObjectAPIName –> eg: Student__Share