**Daily Adjustment Changes:**

Add a Title: Daily CooperSurgical Adjustments

Add the following fields: Adjustment Date, Period, Adjustment Comment

Default Date for Adjustment Date should be yesterday

Make the following fields Editable (but provide a drop down list of available options): Adjustment Date (Use a calendar), Country Name, Company Name, Sub Business Unit, Sub Segment Name, Account Sub Type Name, Sub Category Name

Remove fields: Update User, Update Date, Adjustment Frequency

Do not have rows disappear after data has been entered into the Adjustment Table

If a row is updated after the initial entry, replace the Cost/Quantity/Amount with the new values entered.

Default Values: Adjustment Date = Yesterday’s date. Period = Period ID where the Adjustment Date is between the Period Start Date and Period End Date from the table CSIDW.dbo.Period.

Logic for Period: Once the user has entered the Adjustment Date, set the Period = Period ID in the table CSIDW.dbo.Period where the Adjustment Date is between the Period Start Date and Period End Date.

Additional Logic: All Daily Adjustments should be editable for the Current Open Period and the Next Open Period.

Finding Open Period: SELECT Period FROM Period WHERE OpenPeriodFlag = 1

Current Period is 10, any adjustments entered for period 10 or 11 in fiscal year 2017 should be editable at this point. Once Period 10 is closed, period 11 and 12 should be editable.

SELECT DATEADD(day,DATEDIFF(day,0,GETDATE()),-1) As AdjustmentDate,

h.Period,

a.AdjustmentQuantity,

a.AdjustmentAmount,

a.AdjustmentCost,

b.AdjustmentTypeName,

c.CountryName,

d.SubBusinessUnitName,

g.CompanyName,

e.SegmentName,

f.AccountSubTypeName,

i.SubCategoryName,

'' As AdjustmentComment

FROM AdjustmentTemplate a

INNER JOIN AdjustmentType b ON a.AdjustmentTypeID = b.AdjustmentTypeID

INNER JOIN Country c ON a.CountryID = c.CountryID

INNER JOIN SubBusinessUnit d ON a.SubBusinessUnitID = d.SubBusinessUnitID

INNER JOIN Segment e ON a.SubSegmentID = e.SegmentID

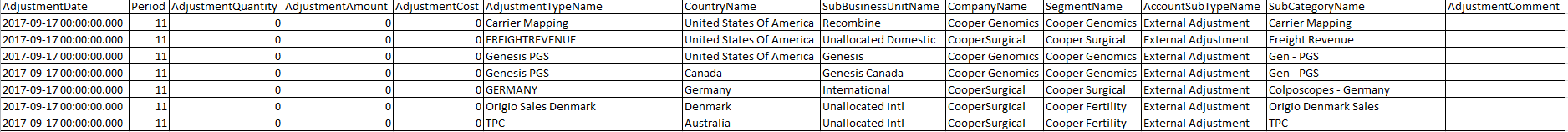
INNER JOIN AccountSubType f ON a.AccountSubTypeID = f.AccountSubTypeID

INNER JOIN Company g ON a.CompanyID = g.CompanyID

INNER JOIN (SELECT Period,PeriodStartDate,PeriodEndDate FROM Period) h ON DATEADD(day,DATEDIFF(day,0,GETDATE()),-1) BETWEEN PeriodStartDate AND PeriodEndDate

INNER JOIN SubCategory i ON a.SubCategoryID = i.SubCategoryID

WHERE a.AdjustmentFrequency = 'D'



Populating Drop Down Lists Logic:

Country Name: After Selecting a Company

SELECT DISTINCT CountryName

FROM Country a

INNER JOIN Company b ON a.CompanyID = b.CompanyID

WHERE CompanyName =

Company Name: All Company Names

SELECT DISTINCT CompanyName

FROM Company

Sub Business Unit: After Selecting a Company

SELECT DISTINCT SubBusinessUnitName

FROM SubBusinessUnit a

INNER JOIN Company b ON a.CompanyID = b.CompanyID

WHERE CompanyName =

Sub Segment: Can be populated the same time As Company

SELECT DISTINCT SegmentName

FROM Company a

INNER JOIN Segment b ON a.SubSegmentID = b.SegmentID

WHERE CompanyName =

Account Sub Type Name:

SELECT DISTINCT AccountSubTypeName

FROM AccountSubType a

INNER JOIN AccountType b ON a.AccountTypeID = b.AccountTypeID

WHERE b.AccountTypeName = 'Adjustment'

Sub Category Name: Only the valid values for the Company/Sub Business Unit. I think I have to add another table for this logic.

**Monthly Adjustment Changes:**

Add a Title: Monthly CooperSurgical Adjustments

Add the following fields: Adjustment Date, Period, Adjustment Comment

Default Date for Adjustment Date should be the last day of the previous month

Make the following fields editable (but provide a drop down list box of available options): Company Name, Country Name, Sub Business Unit Name, Sub Category Name, Sub Segment Name, Adjustment Date (Use a calendar)

Remove Fields: Update User, Update Date, Adjustment Frequency

Show All Monthly Adjustments on 1 page OR do not exit Monthly Adjustment screen when 2 or 3 is clicked on

Default Values: Adjustment Date = SELECT PeriodEndDate FROM Period WHERE OpenPeriodFlag = 1

Period = SELECT Period FROM Period WHERE OpenPeriodFlag = 1

Logic for Period (If Default value is changed only): Period = Period ID in CSIDW.dbo.Period where Adjustment Date = PeriodEndDate

Additional Logic: Add a check box for Reverse Next Month and Duplicate Row default value = FALSE. Do not remove monthly adjustment from screen once data has been entered into Adjustment Table. All monthly adjustments should be editable while the period is open.

SELECT h.PeriodEndDate As AdjustmentDate,

h.Period,

a.AdjustmentQuantity,

a.AdjustmentAmount,

a.AdjustmentCost,

b.AdjustmentTypeName,

c.CountryName,

d.SubBusinessUnitName,

g.CompanyName,

e.SegmentName,

f.AccountSubTypeName,

i.SubCategoryName,

'' As AdjustmentComment

FROM AdjustmentTemplate a

INNER JOIN AdjustmentType b ON a.AdjustmentTypeID = b.AdjustmentTypeID

INNER JOIN Country c ON a.CountryID = c.CountryID

INNER JOIN SubBusinessUnit d ON a.SubBusinessUnitID = d.SubBusinessUnitID

INNER JOIN Segment e ON a.SubSegmentID = e.SegmentID

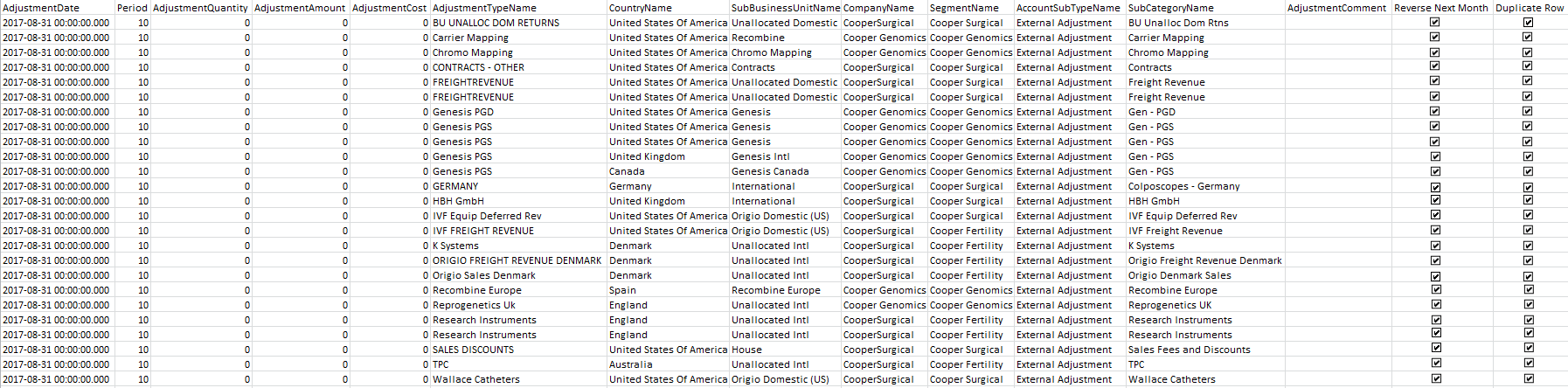
INNER JOIN AccountSubType f ON a.AccountSubTypeID = f.AccountSubTypeID

INNER JOIN Company g ON a.CompanyID = g.CompanyID

INNER JOIN (SELECT Period,PeriodStartDate,PeriodEndDate,OpenPeriodFlag FROM Period) h ON DATEADD(MONTH,DATEDIFF(MONTH,-1,GETDATE())-1,-1) BETWEEN PeriodStartDate AND PeriodEndDate

INNER JOIN SubCategory i ON a.SubCategoryID = i.SubCategoryID

WHERE a.AdjustmentFrequency = 'M' AND h.OpenPeriodFlag = 1



Populating Drop Down Lists Logic:

Country Name: After Selecting a Company

SELECT DISTINCT CountryName

FROM Country a

INNER JOIN Company b ON a.CompanyID = b.CompanyID

WHERE CompanyName =

Company Name: All Company Names

SELECT DISTINCT CompanyName

FROM Company

Sub Business Unit: After Selecting a Company

SELECT DISTINCT SubBusinessUnitName

FROM SubBusinessUnit a

INNER JOIN Company b ON a.CompanyID = b.CompanyID

WHERE CompanyName =

Sub Segment: Can be populated the same time As Company

SELECT DISTINCT SegmentName

FROM Company a

INNER JOIN Segment b ON a.SubSegmentID = b.SegmentID

WHERE CompanyName =

Account Sub Type Name:

SELECT DISTINCT AccountSubTypeName

FROM AccountSubType a

INNER JOIN AccountType b ON a.AccountTypeID = b.AccountTypeID

WHERE b.AccountTypeName = 'Adjustment'

Sub Category Name: Only the valid values for the Company/Sub Business Unit. I think I have to add another table for this logic.

Check Boxes Logic:

Reverse Next Month: If this box is selected do the following

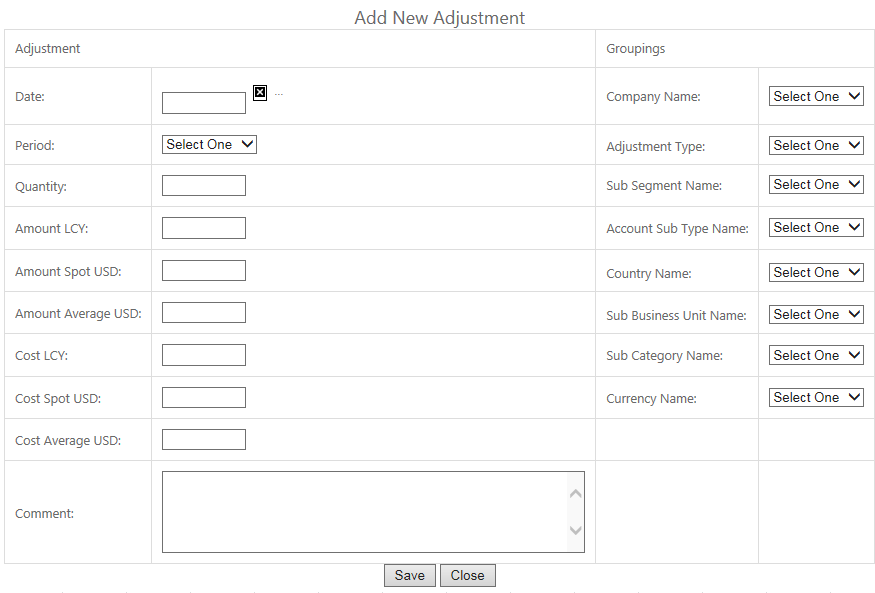
Create n duplicate of the row(s) that is selected but multiple the Amount/Quantity/Cost by -1 with a date of the last day of the next month.

Duplicate Row: If this box is selected do the following

Create an exact duplicate of the row(s) that is selected. The user will then edit the amount, quantity, cost. This should be added as a separate line. DO NOT overwrite the existing line.

**Add New Adjustment Changes:**

Change the order to below



Add the following logic once a company is selected:

* If Company In (CooperSurgical, Cooper Genomics, ReproGenetics) make the following uneditable Amount LCY, Amount Spot USD, Cost LCY, Cost Spot USD. Account Sub Type should default to External Adjustment. Sub Segment Name = Sub Segment in Company table. Currency Name = Company Local Currency in Company table
* If Company Not In (CooperSurgical, Cooper Genomics, ReproGenetics) make the following uneditable Amount Average USD, Amount Spot USD, Cost Average USD, Cost Spot USD. Account Sub Type should default to External Adjustment. Sub Segment Name = Sub Segment in Company table. Currency Name = Company Local Currency in Company table
* Adjustment Type =

SELECT DISTINCT AdjustmentTypeName

FROM AdjustmentType a

INNER JOIN Company b ON a.CompanyID = b.CompanyID

WHERE b.CompanyName =

* Once an Adjustment is selected auto populate Sub Category Name

SELECT DISTINCT AdjustmentTypeName, c.SubCategoryName

FROM AdjustmentType a

INNER JOIN Company b ON a.CompanyID = b.CompanyID

INNER JOIN SubCategory c ON a.SubCategoryID = c.SubCategoryID

WHERE b.CompanyName =

* Country Name =

SELECT DISTINCT CountryName

FROM Country a

INNER JOIN Company b ON a.CompanyID = b.CompanyID

WHERE b.CompanyName =

* Sub Business Unit. I think I have to add another table for this logic.