```
Options macrogen symbolgen mlogic mprint mfile;

/*
     Add ldap_user and ldap_pwd as parameters.

     Change the following 4 macro variables based on the quarterly run.
*/

%let prev_year = 2014;
%let curr_year = 2015;
%let end_month = 9;
%let curr_qtr = 3;

%let pivot_metrics =
AC
AL
AN
AN_HH_N_1
AN_HH_P_0
HH_C
HH_C_D1
HH_C_D1_L2
HH_C_D1_N1
HH_C_D2
HH_L_2
HH_N_1
HH_P_0
;

%let pivot_metrics_yoy =
HH_C_YOY
HH_N_1_YOY
HH_N_1_YOYTOT
;

/*********************************/

libname bmgeth teradata user="&ldap_user@LDAP" password="&ldap_pwd"
tdpid=bmg.wellsfargo.com schema=bmgu_bmgeth;
libname milldata
'/sas/AU48750/BusinessAnalytics/KPMMillennialDashboard/data';

/*
    In step 1, collapse the cube.  Not all attributes of the cube dimensions
are needed for user presentation.  Collapse ethnicities into 6 categories
    and Regional Hierarchy into 4 levels.  Also create a view dimension.
*/

%macro step1;
proc sql noerrorstop;
connect to teradata(server="bmg.wellsfargo.com" user="&ldap_user@LDAP"
password="&ldap_pwd" connection=global mode=teradata);
```

```
execute(
create volatile table mill_mini_cube as
(
select
ASOF_YYYYMM,
(case
when ETH_LEVEL1 in (  'NO ETEK RECORD' , 'UNKNOWN') then '6 - Unknown'
when ETH_LEVEL1 = 'LATINO' then '3 - Hispanic'
when ETH_LEVEL1 in ( 'NATIVE AMERICAN','OTHER') then '4 - Other DS'
when ETH_LEVEL1 = 'AFRICAN AMERICAN' THEN '1 - African American'
when ETH_LEVEL1 = 'ASIAN AMERICAN' THEN '2 - Asian American'
when ETH_LEVEL1 = 'CAUCASIAN' THEN '5 - Caucasian' else ETH_LEVEL1
end)  as Ethnicity,
(case
when gen = 0 then 'Non-Millennial'
when gen = 1 then 'Millennial'
else 'Unknown'
end) as Generation,
case
when SUPER_GRP_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else SUPER_GRP_NM
end as Organization,
case
when GROUP_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else GROUP_NM
end as Lead_Region,
case
when REG_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else REG_NM
end as Region,
case
when TRTRY_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else TRTRY_NM
end as Market,
'Community Bank' as "View",
DOMAIN_LEVEL1,
DOMAIN_LEVEL2,
METRIC,
sum(cnt) as cnt
from bmgu_bmgeth.t_hh_cube_mill

where domain_level1 <> 'LENDING'
and HH_CATEGORY in ('BR','BM')

group by 1,2,3,4,5,6,7,8,9,10,11

union

select
ASOF_YYYYMM,
(case
```

```
when ETH_LEVEL1 in (  'NO ETEK RECORD' , 'UNKNOWN') then '6 - Unknown'
when ETH_LEVEL1 = 'LATINO' then '3 - Hispanic'
when ETH_LEVEL1 in ( 'NATIVE AMERICAN','OTHER') then '4 - Other DS'
when ETH_LEVEL1 = 'AFRICAN AMERICAN' THEN '1 - African American'
when ETH_LEVEL1 = 'ASIAN AMERICAN' THEN '2 - Asian American'
when ETH_LEVEL1 = 'CAUCASIAN' THEN '5 - Caucasian' else ETH_LEVEL1
end)  as Ethnicity,
(case
when gen = 0 then 'Non-Millennial'
when gen = 1 then 'Millennial'
else 'Unknown'
end) as Generation,
case
when SUPER_GRP_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else SUPER_GRP_NM
end as Organization,
case
when GROUP_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else GROUP_NM
end as Lead_Region,
case
when REG_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else REG_NM
end as Region,
case
when TRTRY_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else TRTRY_NM
end as Market,
'All Retail' as "View",
DOMAIN_LEVEL1,
DOMAIN_LEVEL2,
METRIC,
sum(cnt) as cnt
from bmgu_bmgeth.t_hh_cube_mill

where domain_level1 <> 'LENDING'
and HH_CATEGORY in ('BR','BM','NBR','NBM')

group by 1,2,3,4,5,6,7,8,9,10,11

union

select
ASOF_YYYYMM,
(case
when ETH_LEVEL1 in (  'NO ETEK RECORD' , 'UNKNOWN') then '6 - Unknown'
when ETH_LEVEL1 = 'LATINO' then '3 - Hispanic'
when ETH_LEVEL1 in ( 'NATIVE AMERICAN','OTHER') then '4 - Other DS'
when ETH_LEVEL1 = 'AFRICAN AMERICAN' THEN '1 - African American'
when ETH_LEVEL1 = 'ASIAN AMERICAN' THEN '2 - Asian American'
when ETH_LEVEL1 = 'CAUCASIAN' THEN '5 - Caucasian' else ETH_LEVEL1
end)  as Ethnicity,
```

```
(case
when gen = 0 then 'Non-Millennial'
when gen = 1 then 'Millennial'
else 'Unknown'
end) as Generation,
case
when SUPER_GRP_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else SUPER_GRP_NM
end as Organization,
case
when GROUP_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else GROUP_NM
end as Lead_Region,
case
when REG_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else REG_NM
end as Region,
case
when TRTRY_NM = 'STORE_SKEY - NO MATC' THEN 'NON-CB'
else TRTRY_NM
end as Market,
'WIM' as "View",
upper(DOMAIN_LEVEL1) as DOMAIN_LEVEL1,
upper(DOMAIN_LEVEL2) as DOMAIN_LEVEL2,
METRIC,
sum(cnt) as cnt
from bmgu_bmgeth.t_hh_cube_mill

where domain_level1 <> 'LENDING'
and HH_CATEGORY in ('WIM')

group by 1,2,3,4,5,6,7,8,9,10,11
) with data no primary index on commit preserve rows
) by teradata;

/* download volatile table from teradata into SAS data set of same name
*/

create table mill_mini_cube as
select * from connection to teradata
(
select * from mill_mini_cube
);

/* Make sure long enough for derived metric names */

alter table mill_mini_cube modify METRIC char(15);
proc contents data=mill_mini_cube; run;

disconnect from teradata;
quit;
```

```
%mend; /* step1 */

/*
   Step 2:

      Create derived metrics for current year -- YOY, YTD, etc.

*/

%macro step2;

%do i = 1 %to &end_month;

%let prev_yearmonth = %eval(&prev_year * 100 + &I);
%let hh_n_yearmonth = %eval(&prev_year * 100 + &I + 1);
%let curr_yearmonth = %eval(&curr_year * 100 + &I);

/*  HH_C_YOY */

proc sort data=mill_mini_cube out=final_prev(rename=(cnt=cnt_prev));
     where asof_yyyymm = &prev_yearmonth and metric = 'HH_C';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

proc sort data=mill_mini_cube out=final_curr(rename=(cnt=cnt_curr));
     where asof_yyyymm = &curr_yearmonth and metric = 'HH_C';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

data hh_c_yoy_temp(keep=asof_yyyymm Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt metric
cnt_prev cnt_curr);
      length metric $ 15;
     merge final_prev(keep=Ethnicity Generation Organization Lead_Region
Region Market view domain_level1 domain_level2 cnt_prev in=in1)
            final_curr(keep=Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt_curr
in=in2);
      by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
      metric = 'HH_C_YOY';
      asof_yyyymm = &curr_yearmonth;
      if in1 and in2 then cnt = cnt_curr - cnt_prev;
      else if in1 and not in2 then cnt = 0 - cnt_prev;
     else if not in1 and in2 then cnt = cnt_curr;
run;

/* HH_C_YTD */
```

```
proc sort data=mill_mini_cube out=final_prev(rename=(cnt=cnt_prev));
     where asof_yyyymm = &prev_year.13 and metric = 'HH_C';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

data hh_c_ytd_temp(keep=asof_yyyymm Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt metric
cnt_prev cnt_curr);
      length metric $ 15;
     merge final_prev(keep=Ethnicity Generation Organization Lead_Region
Region Market view domain_level1 domain_level2 cnt_prev in=in1)
            final_curr(keep=Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt_curr
in=in2);
      by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
      metric = 'HH_C_YTD';
      asof_yyyymm = &curr_yearmonth;
      if in1 and in2 then cnt = cnt_curr - cnt_prev;
      else if in1 and not in2 then cnt = 0 - cnt_prev;
     else if not in1 and in2 then cnt = cnt_curr;
run;


/*  HH_N_1_YOY */

proc sort data=mill_mini_cube out=final_prev(rename=(cnt=cnt_prev));
     where asof_yyyymm = &prev_yearmonth and metric = 'HH_N_1';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

proc sort data=mill_mini_cube out=final_curr(rename=(cnt=cnt_curr));
     where asof_yyyymm = &curr_yearmonth and metric = 'HH_N_1';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

data hh_n_1_yoy_temp(keep=asof_yyyymm Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt metric
cnt_prev cnt_curr);
      length metric $ 15;
     merge final_prev(keep=Ethnicity Generation Organization Lead_Region
Region Market view domain_level1 domain_level2 cnt_prev in=in1)
            final_curr(keep=Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt_curr
in=in2);
      by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
      metric = 'HH_N_1_YOY';
      asof_yyyymm = &curr_yearmonth;
```

```
      if in1 and in2 then cnt = cnt_curr - cnt_prev;
      else if in1 and not in2 then cnt = 0 - cnt_prev;
     else if not in1 and in2 then cnt = cnt_curr;
run;


/* HH_N_1_YTD */

proc sort data=mill_mini_cube out=final_prev(rename=(cnt=cnt_prev));
     where asof_yyyymm = &prev_year.13 and metric = 'HH_N_1';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

data hh_n_1_ytd_temp(keep=asof_yyyymm Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt metric
cnt_prev cnt_curr);
      length metric $ 15;
     merge final_prev(keep=Ethnicity Generation Organization Lead_Region
Region Market view domain_level1 domain_level2 cnt_prev in=in1)
           final_curr(keep=Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt_curr
in=in2);
      by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
      asof_yyyymm = &curr_yearmonth;
      metric = 'HH_N_1_YTD';
      if in1 and in2 then cnt = cnt_curr - cnt_prev;
      else if in1 and not in2 then cnt = 0 - cnt_prev;
     else if not in1 and in2 then cnt = cnt_curr;
run;


/* HH_N_1_YOYTOT (12 month total) */

proc sql noerrorstop;
create table hh_n_1_yoytot_temp as
select
&curr_yearmonth as ASOF_YYYYMM,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
Market,
view,
domain_level1,
domain_level2,
'HH_N_1_YOYTOT' as metric length=15,
sum(case when metric = 'HH_N_1' and a.asof_yyyymm between &hh_n_yearmonth
and &curr_yearmonth then cnt else 0 end) as cnt
from mill_mini_cube a
```

```
where a.ASOF_YYYYMM ne 201413 /* exclude month13 */

group by 1,2,3,4,5,6,7,8,9,10
;
quit;


/*  AC_YOY */

proc sort data=mill_mini_cube out=final_prev(rename=(cnt=cnt_prev));
     where asof_yyyymm = &prev_yearmonth and metric = 'AC';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

proc sort data=mill_mini_cube out=final_curr(rename=(cnt=cnt_curr));
     where asof_yyyymm = &curr_yearmonth and metric = 'AC';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

data ac_yoy_temp(keep=asof_yyyymm Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt metric
cnt_prev cnt_curr);
       length metric $ 15;
     merge final_prev(keep=Ethnicity Generation Organization Lead_Region
Region Market view domain_level1 domain_level2 cnt_prev in=in1)
             final_curr(keep=Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt_curr
in=in2);
       by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
       metric = 'AC_YOY';
       asof_yyyymm = &curr_yearmonth;
       if in1 and in2 then cnt = cnt_curr - cnt_prev;
       else if in1 and not in2 then cnt = 0 - cnt_prev;
     else if not in1 and in2 then cnt = cnt_curr;
run;


/* AC_YTD */

proc sort data=mill_mini_cube out=final_prev(rename=(cnt=cnt_prev));
     where asof_yyyymm = &prev_year.13 and metric = 'AC';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

data ac_ytd_temp(keep=asof_yyyymm Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt metric
cnt_prev cnt_curr);
```

```
      length metric $ 15;
     merge final_prev(keep=Ethnicity Generation Organization Lead_Region
Region Market view domain_level1 domain_level2 cnt_prev in=in1)
            final_curr(keep=Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt_curr
in=in2);
      by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
      metric = 'AC_YTD';
      asof_yyyymm = &curr_yearmonth;
      if in1 and in2 then cnt = cnt_curr - cnt_prev;
      else if in1 and not in2 then cnt = 0 - cnt_prev;
     else if not in1 and in2 then cnt = cnt_curr;
run;


/*  AN_YOY */

proc sort data=mill_mini_cube out=final_prev(rename=(cnt=cnt_prev));
     where asof_yyyymm = &prev_yearmonth and metric = 'AN';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

proc sort data=mill_mini_cube out=final_curr(rename=(cnt=cnt_curr));
     where asof_yyyymm = &curr_yearmonth and metric = 'AN';
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

data an_yoy_temp(keep=asof_yyyymm Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt metric
cnt_prev cnt_curr);
      length metric $ 15;
     merge final_prev(keep=Ethnicity Generation Organization Lead_Region
Region Market view domain_level1 domain_level2 cnt_prev in=in1)
            final_curr(keep=Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt_curr
in=in2);
      by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
      metric = 'AN_YOY';
      asof_yyyymm = &curr_yearmonth;
      if in1 and in2 then cnt = cnt_curr - cnt_prev;
      else if in1 and not in2 then cnt = 0 - cnt_prev;
     else if not in1 and in2 then cnt = cnt_curr;
run;

/* AN_YTD */

proc sort data=mill_mini_cube out=final_prev(rename=(cnt=cnt_prev));
     where asof_yyyymm = &prev_year.13 and metric = 'AN';
```

```
     by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
run;

data an_ytd_temp(keep=asof_yyyymm Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt metric
cnt_prev cnt_curr);
       length metric $ 15;
     merge final_prev(keep=Ethnicity Generation Organization Lead_Region
Region Market view domain_level1 domain_level2 cnt_prev in=in1)
            final_curr(keep=Ethnicity Generation Organization
Lead_Region Region Market view domain_level1 domain_level2 cnt_curr
in=in2);
       by Ethnicity Generation Organization Lead_Region Region Market view
domain_level1 domain_level2;
       metric = 'AN_YTD';
       asof_yyyymm = &curr_yearmonth;
       if in1 and in2 then cnt = cnt_curr - cnt_prev;
       else if in1 and not in2 then cnt = 0 - cnt_prev;
     else if not in1 and in2 then cnt = cnt_curr;
run;

/* AN_YOYTOT (12 month total) */

proc sql noerrorstop;
create table an_yoytot_temp as
select
&curr_yearmonth as ASOF_YYYYMM,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
Market,
view,
domain_level1,
domain_level2,
'AN_YOYTOT' as metric length=15,
sum(case when metric = 'AN' and a.asof_yyyymm between &hh_n_yearmonth and
&curr_yearmonth then cnt else 0 end) as cnt
from mill_mini_cube a

where a.ASOF_YYYYMM ne 201413 /* exclude month13 */

group by 1,2,3,4,5,6,7,8,9,10
;
quit;

data derived_metrics;
     set %if &i ne 1 %then derived_metrics;
          hh_c_yoy_temp hh_c_ytd_temp hh_n_1_yoy_temp hh_n_1_ytd_temp
ac_yoy_temp ac_ytd_temp an_yoy_temp an_ytd_temp an_yoytot_temp
```

```
hh_n_1_yoytot_temp;
run;


/* drop monthly dataset no longer needed */

proc datasets lib=work nolist;
delete hh_c_yoy_temp hh_c_ytd_temp hh_n_1_yoy_temp hh_n_1_ytd_temp
ac_yoy_temp ac_ytd_temp an_yoy_temp an_ytd_temp an_yoytot_temp
hh_n_1_yoytot_temp;
quit;
run;

%end;

/* Full dataset used as input for tableau back-end and pivot tables */

data mill_mini_cube;
     set derived_metrics(drop=cnt_prev cnt_curr) mill_mini_cube;
run;

/* derived metrics no longer needed */

proc datasets lib=work nolist;
     delete derived_metrics final_prev final_curr;
quit;
run;

%mend; /* step 2 */

/*
    Step 3:

       Create a dataset for Tableau staging.
*/

%macro step3;

proc sql noerrorstop;

create table milldata.tableau as

/* All derived metrics */

select
*
from mill_mini_cube
where metric in (
'HH_C_YOY',
'HH_C_YTD',
'HH_N_1_YOY',
'HH_N_1_YTD',
```

```
'AC_YOY',
'AC_YTD',
'AN_YOY',
'AN_YTD',
'AN_YOYTOT'
'HH_N_1_YOYTOT'
)

union

/* Metrics that can be display for prior 4 months */

select
*
from mill_mini_cube

where metric in ('XS','HH_C','HH_P_0','HH_L_2','HH_N_1')
and asof_yyyymm ge &prev_year.10
and asof_yyyymm lt &prev_year.13

union

/* Metrics for month 13 */

select
*
from mill_mini_cube

where metric in ('HH_C','AC','AL','AN')
and asof_yyyymm eq &prev_year.13

union

/* Metrics for current year */

select
*
from mill_mini_cube

where metric in
('AC','AL','AN','HH_C_D1','HH_C_D2','XS','HH_C','HH_P_0','HH_L_2','HH_
N_1')
and asof_yyyymm gt &prev_year.13
;


quit;


%mend; /* step 3 */

/*
```

```
    Step 4:

   Create excel files that will be used as the data for the pivot tables
*/

%macro step4;


proc sql noerrorstop;

/* To fit into excel, don't include Market and collapse generation */

create table mill_mini_cube_pivot as
select
ASOF_YYYYMM,
Ethnicity,
(case
when Generation = 'Millennial' then 'Millennial    '
else 'Non-Millennial'
end) as Generation,
Organization,
Lead_Region,
Region,
View,
domain_level1,
domain_level2,
sum(cnt) as cnt,
metric
from mill_mini_cube

group by 1,2,3,4,5,6,7,8,9,11
order by 1,2,3,4,5,6,7,8,9,11
;
quit;


%let i=1;
%let metric=%scan(&pivot_metrics,&i);
%do %until (&metric = %nrstr());

proc sql noerrorstop;

create table &metric._&prev_year as
select
asof_yyyymm,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
View,
domain_level1,
```

```
domain_level2,
cnt
from mill_mini_cube_pivot

where asof_yyyymm < &prev_year.13
and metric = "&metric"
;


create table &metric._&curr_year as
select
asof_yyyymm,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
View,
domain_level1,
domain_level2,
cnt
from mill_mini_cube_pivot

where asof_yyyymm >= &prev_year.13
and metric = "&metric"
;

quit;

proc export
     data=&metric._&prev_year
      dblabel
     dbms=xlsx

outfile="/sas/AU48750/BusinessAnalytics/KPMMillennialDashboard/data/&m
etric._&prev_year..xlsx"
      replace;
run;


proc export
     data=&metric._&curr_year
      dblabel
     dbms=xlsx

outfile="/sas/AU48750/BusinessAnalytics/KPMMillennialDashboard/data/&m
etric._&curr_year..xlsx"
      replace;
run;

%put i = &i;
%put metric = &metric;
```

```
%let i = %eval(&i+1);
%let metric=%scan(&pivot_metrics,&i);
%end;



/* Create pivots for YOY metrics */

%let i=1;
%let metric=%scan(&pivot_metrics_yoy,&i);
%do %until (&metric = %nrstr());

proc sql noerrorstop;

create table &metric._&curr_year as
select
asof_yyyymm,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
View,
domain_level1,
domain_level2,
cnt
from mill_mini_cube_pivot

where metric = "&metric"
;

quit;

proc export
     data=&metric._&curr_year
      dblabel
     dbms=xlsx

outfile="/sas/AU48750/BusinessAnalytics/KPMMillennialDashboard/data/&m
etric._&curr_year..xlsx"
      replace;
run;

%let i = %eval(&i+1);
%let metric=%scan(&pivot_metrics_yoy,&i);
%end;


/* Product Penetration - in pivot will be calculated field of cnt/hh_cnt
*/
```

```
proc sql noerrorstop;


create table pp as
select
x.ASOF_YYYYMM,
x.Ethnicity,
x.Generation,
x.Organization,
x.Lead_Region,
x.Region,
x.View,
x.domain_level1,
x.domain_level2,
x.cnt as cnt,
y.cnt as hh_cnt,
'PP' as metric
from(
select
ASOF_YYYYMM,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
View,
DOMAIN_LEVEL1,
DOMAIN_LEVEL2,
sum(cnt) as cnt
from mill_mini_cube_pivot

where metric = 'HH_C_D2'

group by 1,2,3,4,5,6,7,8,9

union

select
ASOF_YYYYMM,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
View,
DOMAIN_LEVEL1,
'Total Product Type' as DOMAIN_LEVEL2,
sum(cnt) as cnt
from mill_mini_cube_pivot

where metric = 'HH_C_D1'
```

```
group by 1,2,3,4,5,6,7,8,9) x,
(select
ASOF_YYYYMM,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
View,
sum(cnt) as cnt
from mill_mini_cube_pivot

where metric = 'HH_C'

group by 1,2,3,4,5,6,7)  y

where
x.ASOF_YYYYMM = y.ASOF_YYYYMM and
x.Ethnicity = y.Ethnicity and
x.Generation = y.Generation and
x.Organization = y.Organization and
x.Lead_Region = y.Lead_Region and
x.Region = y.Region and
x.View = y.View

order by 1,2,3,4,5,6,7,8,9,12
;

create table pp_&prev_year as
select
asof_yyyymm,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
View,
domain_level1,
domain_level2,
cnt,
hh_cnt
from pp

where asof_yyyymm < &prev_year.13
;


create table pp_&curr_year as
select
asof_yyyymm,
Ethnicity,
Generation,
```

```
Organization,
Lead_Region,
Region,
View,
domain_level1,
domain_level2,
cnt,
hh_cnt
from pp

where asof_yyyymm >= &prev_year.13
;




/* Cross Sell */

create table hh_c as
select
ASOF_YYYYMM,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
View,
cnt as hh_cnt
from mill_mini_cube_pivot

where metric='HH_C'

order by 1,2,3,4,5,6,7;

/*
   In pivot Cross sell will be calculated field XS/HH_C.
*/

create table xs as
select
a.ASOF_YYYYMM,
a.Ethnicity,
a.Generation,
a.Organization,
a.Lead_Region,
a.Region,
a.View,
a.DOMAIN_LEVEL1,
a.DOMAIN_LEVEL2,
a.cnt,
b.hh_cnt,
```

```
a.metric
from mill_mini_cube_pivot a

join hh_c b
on a.ASOF_YYYYMM = b.ASOF_YYYYMM
and a.Ethnicity = b.Ethnicity
and a.Generation = b.Generation
and a.Organization = b.Organization
and a.Lead_Region = b.Lead_Region
and a.Region = b.Region
and a.View = b.View

where a.metric='XS'
;

create table xs_&prev_year as
select
asof_yyyymm,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
View,
domain_level1,
domain_level2,
cnt,
hh_cnt
from xs

where asof_yyyymm < &prev_year.13
;



create table xs_&curr_year as
select
asof_yyyymm,
Ethnicity,
Generation,
Organization,
Lead_Region,
Region,
View,
domain_level1,
domain_level2,
cnt,
hh_cnt
from xs

where asof_yyyymm >= &prev_year.13
;
```

```
quit;

proc export
     data=pp_&prev_year
      dblabel
     dbms=xlsx

outfile="/sas/AU48750/BusinessAnalytics/KPMMillennialDashboard/data/pp
_&prev_year..xlsx"
      replace;
run;


proc export
     data=pp_&curr_year
      dblabel
     dbms=xlsx

outfile="/sas/AU48750/BusinessAnalytics/KPMMillennialDashboard/data/pp
_&curr_year..xlsx"
      replace;
run;

proc export
     data=xs_&curr_year
      dblabel
     dbms=xlsx

outfile="/sas/AU48750/BusinessAnalytics/KPMMillennialDashboard/data/xs
_&curr_year..xlsx"
      replace;
run;
proc export
     data=xs_&prev_year
      dblabel
     dbms=xlsx

outfile="/sas/AU48750/BusinessAnalytics/KPMMillennialDashboard/data/xs
_&prev_year..xlsx"
      replace;
run;

%mend; /* step 4 */

%put timestamp = %sysfunc(time(),time8.0);
%step1;
%put timestamp = %sysfunc(time(),time8.0);
%step2;
%put timestamp = %sysfunc(time(),time8.0);
%step3;
%put timestamp = %sysfunc(time(),time8.0);
```

```
/*%step4;*/
%put timestamp = %sysfunc(time(),time8.0);
```