**Visualization Rewrite notes**

Tables required (other than standard EC output). Highlighted tables need to be updated with each run.

1. mom\_baby
2. GCRG\_DETAIL.csv
3. KPMG\_CRG\_DATA\_CY12\_13.txt
4. hiv\_recip
5. dd\_recip
6. harp\_recip
7. mltc\_recip
8. PPS\_table
9. MCO\_table
10. HH\_table
11. PCP\_table
12. ny\_zip\_to\_county
13. county\_mcregion
14. MMCOR\_provider\_type? – Does this need to be updated?

Run order

1. Enrollment\_month –
   1. creates enrolled month table. How is currently works is if the first day of the month the patient is enrolled they are counted as enrolled. Do we want to take into account how many days during the month a patient is enrolled, if so what number of days do they need to be enrolled to be counted as having enrollment for that month?
2. Crg.sql
3. Crg\_summary
   1. Move Crg\_summary into script 5, need TCGP w/o maternity costs and member\_subpopulation2 which are created in 5
4. Claim Type cost roll ups
   1. Programs 4a – 4f
   2. Created claim type cost roll ups for all cost combinations
5. Maternity Bundle
   1. Brings pregnancy, delivery and newborn information on one line for all moms with a record in the mom\_baby file
      1. Multiple are excluded as well as babies assigned to 2 moms (very few but did happen)
6. JS visualization query rewrite
   * + 1. Build episode reference
          1. Creates new build episode reference table. In the past I found we had an old table and we want to ensure we are using the most current 1. I will check the table before running this step and will only run if I find old data in the table.

Checks

Is end\_of\_study set to 1 for ES9901?

Is Fusion episode included?

* + - 1. Create Provider Prep
         1. Takes select fields from the provider table and join on the provider type from MMCOR\_provider\_type table
      2. Create Pac Groups
         1. Creates a list of all pac codes for each episode. This was needed before but is no longer used. I am keeping it in here just in case we end up needing this again.
      3. Create PAC\_groups\_unique
         1. Creates an unduplicated list of all PAC codes in the metadata
         2. This was used for TCGP pacs at one point but then we switched from this method. I am keeping the table just in case we switch back.
      4. Create member\_sub\_population\_2
         1. Creates a member demographic table with the member\_id, subpopulation (HIV,HARP,DD,MLTC,GENERAL), gender, zip\_code, age\_group, county, mcregion, PPS, MCP, HH and PCP.
      5. Create member\_sub\_population2\_exclusive
         1. Create a member demographic table using member\_sub\_population2 implementing the following hierarchy

'HARP & HIV'

‘HIV’

‘HARP’

‘DD’

‘MLTC’

* + - 1. Create co\_occurance\_of\_chronic\_episodes
         1. Creates table with 1 row for each episode and a flag for all chronic episodes occurring at the same time
      2. Create table member\_vistualization\_claim\_type\_temp
         1. joins claims combined and assignment to create a table with 1 record per claim with a maternity flag added for exclusion from TCGP and PPR/PPV flags added for calculation costs.
      3. Create member\_vistualization\_claim\_typea
         1. Creates a table with 1 record per member per year using the marernity flag to exclude maternity costs in where statement
      4. Create table member\_vistualization\_claim\_type (ADD RECORD FOR MEMBERS WITH0 COSTS)
         1. Joins member\_vistualization\_claim\_typea with PAC costs for TCGP and member demographic info
         2. PAC costs for TCGP calculation(ADD DEPRESSION REMOVE ADHD, REMOVE SUDS?)

level=5 and split=1 and annualized=1 and filter\_id=1 and left(master\_epid\_level.master\_episode\_id,6) in ('ES9901','EC0801','EC0802','EC1001','EC1903','EC1905','EC1906','EC1908','EC1910','EC0401','EC0402','EC0508','EC0511','EC0518','EC0521','EC0601')

* + - 1. Create table exp\_cost\_qacrg3\_age\_gender
         1. Creates TCGP, PPR and PPV expected costs
         2. TCGP expected cost calculation

sum(ip\_cost+ op\_cost+ pb\_cost+rx\_cost)/ COUNT(distinct cc.member\_id)

WHERE cc.member\_population NOT IN ('MLTC', 'DD') and enrolled\_month.enrolled\_month=12

GROUP BY

crg.year,

crg.fincrg,

cms\_age\_group,

gender

* + - * 1. PPR and PPV expected costs use same group by and where statement. The cost field is the sum of the PPR/PPV costs calculated in the member\_vistualization\_claim\_type table
      1. Create crg\_cost\_summary\_by\_member
         1. Table with excepted costs from the exp\_cost\_qacrg3\_age\_gender and actual costs from the member\_vistualization\_claim\_type
      2. The next set of queries takes the tables above and updates the report\_episode\_detail table to include the relevant addition fields
      3. Create visual\_analysis\_table\_js (ADD EXCLUSION TO EXCLUDE ZERO DOLLAR ANY CHRONIC OR SICK CARE EPISODES )( Sick and prevent expected costs should be the average cost)( If negative expected costs set to 0 for chronic episodes)
         1. First the episode level records are filled in for episodes which pass filtering

Annualized costs are populated where end\_of\_study=1 else total split costs are used (from RED table)

Demographic information taken from member\_sub\_population2

Utilization field calculation (NOTE: since we switched the episode level ip/op/pb/rx costs to represent the split of the costs for those claim types assigned to the episode this will be representative of the members utilization within the episode, is this ok?)( Needs to be based on total cost and not within episode costs)

case when (case when member\_vistualization\_claim\_type.ip\_cost is null then 0 else member\_vistualization\_claim\_type.ip\_cost end )+

(case when member\_vistualization\_claim\_type.op\_cost is null then 0 else member\_vistualization\_claim\_type.op\_cost end )+

(case when member\_vistualization\_claim\_type.pb\_cost is null then 0 else member\_vistualization\_claim\_type.pb\_cost end )+

(case when member\_vistualization\_claim\_type.rx\_cost is null then 0 else member\_vistualization\_claim\_type.rx\_cost end )<200 then 'Low' else 'Non-Low' end AS utilization,

vbp\_arrangement field created

case when member\_sub\_population2.sub\_population = 'General' AND report\_episode\_detail.episode\_name IN ('URI', 'RHNTS', 'PREVNT', 'SICKCR', 'TONSIL', 'IPC')then 'Integrated Primary Care'

when member\_sub\_population2.sub\_population = 'General' AND report\_episode\_detail.episode\_name IN ('PREGN','CSECT', 'VAGDEL', 'NBORN','MTRNTY')then 'Maternity Care'

when member\_sub\_population2.sub\_population = 'General' AND report\_episode\_detail.episode\_type = 'Chronic' AND report\_episode\_detail. episode\_name NOT IN ('RHNTS', 'DIVERT', 'ULCLTS', 'GLCOMA', 'SCHIZO', 'CROHNS' ,'ADHD')then 'Chronic Bundle' else null end

as vbp\_arrangement

* + - 1. Create visual\_analysis\_table\_js chronic bundle records
         1. Cost fields are a sum of the following where and group by statements

WHERE

r.analysis\_type = 'Episode'

AND r.episode\_type='Chronic'

AND r.episode\_name not in ('RHNTS','DIVERT', 'ULCLTS', 'GLCOMA', 'SCHIZO', 'CROHNS','ADHD')

AND r.episode\_level = 5

GROUP BY

r.member\_id,

r.member\_population,

r.member\_region,

r.subgroup,

r.exclusive,

r.Year

* + - 1. Create visual\_analysis\_table\_js IPC records
         1. Cost fields are a sum of the following where and group by statements

WHERE

r.analysis\_type = 'Episode'

AND r.episode\_name in ('PREVNT', 'SICKCR','Chronic bundle' )

AND r.episode\_level = 5

GROUP BY

r.member\_id,

r.member\_population,

r.member\_region,

r.subgroup,

r.exclusive,

r.Year

* + - 1. Create visual\_analysis\_table\_js Maternity bundle records (need to update code, right now the way the code is working only the pregnancy costs with be in the IP/op/pb/rx within episode split, I need to add in the delivery and baby costs.) (EXCLUDE MATERNITY EPISODES THAT ARE NOT IN THE MOM\_BABY FILE)
         1. Cost fields are created using the following statements

sum(MATERNITY\_BUNDLE\_COSTFIELDS.SPLIT\_TOTAL\_PREGN\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.SPLIT\_TOTAL\_DEL\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.SPLIT\_TOTAL\_NEWBORN\_COST) as Split\_Total\_Cost,

sum(MATERNITY\_BUNDLE\_COSTFIELDS.SPLIT\_PAC\_PREGN\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.SPLIT\_PAC\_DEL\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.SPLIT\_PAC\_NEWBORN\_COST )as Split\_Total\_PAC\_Cost,

sum(MATERNITY\_BUNDLE\_COSTFIELDS.SPLIT\_TYPICAL\_PREGN\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.SPLIT\_TYPICAL\_DEL\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.SPLIT\_TYPICAL\_NEWBORN\_COST) as Split\_Total\_Typical\_Cost,

sum(MATERNITY\_BUNDLE\_COSTFIELDS. UNSPLIT\_TOTAL\_PREGN\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.UNSPLIT\_TOTAL\_DEL\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.UNSPLIT\_TOTAL\_NEWBORN\_COST) as Unsplit\_Total\_Cost,

sum(MATERNITY\_BUNDLE\_COSTFIELDS.UNSPLIT\_PAC\_PREGN\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.UNSPLIT\_PAC\_DEL\_COST+ MATERNITY\_BUNDLE\_COSTFIELDS.UNSPLIT\_PAC\_NEWBORN\_COST ) as Unsplit\_Total\_PAC\_Cost,

sum(MATERNITY\_BUNDLE\_COSTFIELDS.UNSPLIT\_TYPICAL\_PREGN\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.UNSPLIT\_TYPICAL\_DEL\_COST + MATERNITY\_BUNDLE\_COSTFIELDS.UNSPLIT\_TYPICAL\_NEWBORN\_COST) as Unsplit\_Total\_Typical\_Cost,

PREG\_AVE\_SPLIT\_COSTS+ split\_expected\_del\_total\_cost + NEWBORN\_AVE\_SPLIT\_COSTS as Split\_Expected\_Total\_Cost,

(PREG\_AVE\_SPLIT\_COSTS+ split\_expected\_del\_total\_cost + NEWBORN\_AVE\_SPLIT\_COSTS)-(PREG\_AVE\_SPLIT\_PAC\_COSTS+ split\_expected\_del\_pac\_cost + NEWBORN\_AVE\_SPLIT\_PAC\_COSTS) as Split\_Expected\_Typical\_Other\_Cost,

PREG\_AVE\_SPLIT\_PAC\_COSTS+ split\_expected\_del\_pac\_cost +NEWBORN\_AVE\_SPLIT\_PAC\_COSTS as Split\_Expected\_PAC\_Cost,

PREG\_AVE\_UNSPLIT\_COSTS+ unsplit\_expected\_del\_total\_cost + NEWBORN\_AVE\_UNSPLIT\_COSTS as Unsplit\_Expected\_Total\_Cost,

(PREG\_AVE\_UNSPLIT\_COSTS+ unsplit\_expected\_del\_total\_cost + NEWBORN\_AVE\_UNSPLIT\_COSTS)- (PREG\_AVE\_UNSPLIT\_PAC\_COSTS+ unsplit\_expected\_del\_pac\_cost +NEWBORN\_AVE\_UNSPLIT\_PAC\_COSTS) as

Unsplit\_Expected\_Typical\_Other\_Cost,

PREG\_AVE\_UNSPLIT\_PAC\_COSTS+ unsplit\_expected\_del\_pac\_cost +NEWBORN\_AVE\_UNSPLIT\_PAC\_COSTS as Unsplit\_Expected\_PAC\_Cost

* + - 1. Create visual\_analysis\_table\_js Population (TCGP) level records(Look for negatives. If pac expected higher than total expected make typical 0 and pac=total expected)
         1. Costs field calculated as follows

(member\_vistualization\_claim\_type.ip\_cost + member\_vistualization\_claim\_type.op\_cost +member\_vistualization\_claim\_type.pb\_cost +member\_vistualization\_claim\_type.rx\_cost) as Unsplit\_Total\_Cost,

(member\_vistualization\_claim\_type.ip\_cost + member\_vistualization\_claim\_type.op\_cost +member\_vistualization\_claim\_type.pb\_cost +member\_vistualization\_claim\_type.rx\_cost)- member\_vistualization\_claim\_type.pac\_cost as Unsplit\_Total\_Typical\_Cost,

(crg\_cost\_summary\_by\_member.expected\_cost) as Unsplit\_Expected\_Total\_Cost,

(crg\_cost\_summary\_by\_member.expected\_cost)-(sum(case wher.analysis\_type = 'Episode' AND r.episode\_name ='Chronic bundle' AND r.episode\_level = 5 then r.Split\_Expected\_PAC\_Cost else 0 end)+sum(case wher.analysis\_type = 'Episode' AND r.episode\_name ='SICKCR' AND r.episode\_level = 5 then r. Split\_Total\_PAC\_Cost else 0 end) ) as Unsplit\_Expected\_Typical\_Other\_Cost

sum(case wher.analysis\_type = 'Episode' AND r.episode\_name ='Chronic bundle' AND r.episode\_level = 5 then r.Split\_Expected\_PAC\_Cost else 0 end)+sum(case wher.analysis\_type = 'Episode' AND r.episode\_name ='SICKCR' AND r.episode\_level = 5 then r. Split\_Total\_PAC\_Cost else 0 end) as Unsplit\_Expected\_PAC\_Cost,

where r.enrolled\_num\_month = 12

* + - 1. Create table epi\_counts
         1. Create counts of episode for the TCGP rows, both filtered and unfiltered getting a distinct count of master\_episode\_id by filter\_id by year
      2. Create table temp\_main\_state\_wide\_values\_2
         1. Creates averages using the follow group by and where statements
         2. WHERE

member\_county IS NOT NULL

* + - * 1. Group by

analysis\_type, vbp\_arrangement, member\_population, episode\_name, year, episode\_level

* + - 1. Update visual\_analysis\_table\_va\_js to include vbp\_contractor and pcp
      2. Create visual\_analysis\_table\_js\_subset
         1. Select only episodes needed for tableau reports
      3. Update visual\_analysis\_table\_js\_subset to fill in crg fields from crg table based on member\_id and year