****

**CHIA Enrollment Trends (February 2018 Edition)**

**MA APCD Programming Code**

As described in the CHIA Enrollment Trends Technical Appendix, Enrollment Trends monitors unique Massachusetts residents with primary medical insurance coverage using eligibility data from the Massachusetts All Payer Claims Database (MA APCD). MA APCD membership reports are generated on a bi-annual basis and sourced from the most recently submitted payer-specific Member Eligibility (ME) file. Twenty-four months of membership data are used and segmented by product enrollment date spans.

Provided below are the methodology and programming code used by CHIA to create the MA APCD membership reports for Enrollment Trends. The methodology and code have been discussed with and reviewed by the top 13 commercial payers[[1]](#footnote-1) (and affiliated entities) but are subject to further development in order to arrive at the most accurate membership counts for each payer. On a case-by-case basis, CHIA may request supplemental data from payers to augment or replace MA APCD enrollment counts.

1. **Methodology**

Enrollment is determined if the 15th day of the last month of each quarter (i.e., Snapshot Date) is within the Product Enrollment Start Date (ME041) and Product Enrollment End Date (ME042) of a member record (i.e., ME041 ≤ Snapshot Date ≤ ME042). A member record with NULL values for either ME041 or ME042 is considered actively enrolled. Members are distinguished using data element HashCarrierSpecificUniqueMemberID (ME107) by OrgID (ME001).

MA APCD membership reports do not count duplicate member records. A unique member record is selected according to the following criteria, in order of significance:

Subset by the following:

1. Massachusetts resident (ME016=MA)
2. Medical coverage (ME018=1)
3. Primary insurance[[2]](#footnote-2) (ME028=1)

De-duplicate by the following order:

1. Coverage type[[3]](#footnote-3) (ME029)
	1. ME029=(ASO, ASW) for Self-Insured
	2. ME029=(UND) for Fully-Insured
	3. ME029=(STN, OTH) for Other
2. Last activity date (ME056)
3. Member Eligibility ID (Derived-ME05)
4. **Programming Code[[4]](#footnote-4)**

MA APCD data extraction and aggregation are implemented in SAS, using SQL pass-through to access data in Netezza. The code below may not be inclusive of all data manipulations performed on the final output dataset.

1. **SAS/SQL Code**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

 SAS/SQL CODE TO EXTRACT AGGREGATE APCD MEMBERSHIP DATA FOR ENROLLMENT TRENDS REPORTING

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*--------------- BEGIN ACTION ITEMS -------------------------------------------

 1. Initialize global variables

 2. Perform additional action items below prior to running program

--------------- END ACTION ITEMS ---------------------------------------------;

\* Initialize global variables;

%let DataDir= C:\EnrollmentTrends\output;

%let SYM= 201709;

%let OrgIdList= (290, 291, 295, 296, 300, 301, 302, 312, 3505, 3735, 4962, 7655, 7789, 8026, 8647, 9913, 10353, 10441, 10444, 10632, 10647, 10920, 10926, 10929, 11474, 11726, 12226, 12814, 13074);

%let Snapshot1= '2015-12-15'; \*First snapshot date, format: 'yyyy-mm-dd';

%let nMonths= 3; \*# of months to increment snapshot dates, Default=3 (e.g. nMonths=1 means monthly, nMonths=3 means quarterly);

%let nTimeSpan= 8; \*# of time spans, Default=8 (e.g. if n=3 and nTimeSpan=8, then program will generate a snapshot date every 3 months, i.e. every quarter, for 8 quarters);

\* Fixed global variables;

libname out "&DataDir.";

%let OutFile= out.ME\_ET\_&SYM.;

%let StartYear= %substr(&snapshot1,2,4); \*Year of first snapshot date;

%let StartMonth= %substr(&snapshot1,7,2); \*Month of first snapshot date;

%let StartDay= %substr(&snapshot1,10,2); \*Day of first snapshot date;

option compress=binary;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Begin: Netezza summary data extract

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*--------------- BEGIN ACTION ITEMS -------------------------------------------

 Perform the checks below prior to running code

 1. Review "Processing Summary" below for previous processing info

 2. Check that source data is coming from correct data source

--------------- END ACTION ITEMS ---------------------------------------------;

\* Processing Summary:

 As of 11/16/2016: Changing snapshot date to the 15th of the month.

Extract 8 quarters, starting with snapshot date of 06/15/2015, of ME data for Enrollment Trends. The below dataset has the following characteristics:

 1) The dataset has been de-duplicated (to\_keep\_v1= 1)

 2) The dataset contains only OrgIDs relevant to Enrollment Trends

 3) The dataset contains three Product fields: ProductID, ProductLineofBusinessModel, Situs

 4) The dataset contains demographic fields: AgeGroup, County, and MemberGender

;

**proc** **sql**;

 connect to odbc (dsn=odbc\_database);

 create table &OutFile. as

 select \* from connection to odbc (

 with DateList as

 (select add\_months(cast(&startYear.&startMonth.&startDay. as datetime), &nMonths.\*idx) as dates

 from (select &Snapshot1. :: date start\_date) starting\_date

 cross join \_v\_vector\_idx

 where idx < &nTimeSpan.

 ),

 Product as

 (select OrgID

 , HashOrgIDProductIDNumber

 , ProductBenefitType

 , ProductLineofBusinessModel

 , Situs

 , NaicCode

 , CarrierLicenseType

 , ProductName

 , InsurancePlanMarketCleaned

 from (select \*

 , row\_number() over (partition by OrgID, HashOrgIDProductIDNumber order by SubmissionYearMonth desc) as S

 from APCD\_Product) as PR

 where S=**1**

 )

 select OrgID

 , SubmissionControlID

 , SubmissionYearMonth

 , Dates

 , InsuranceTypeCodeProduct

 , MedicalCoverage

 , PrimaryInsuranceIndicator

 , CoverageType

 , MarketCategoryCode

 , SpecialCoverage

 , ProductID

 , ProductLineOfBusinessModel

 , Situs

 , RACP

 , MemberMA

 , County

 , MemberGender

 , AgeGroup

 , PurchasedThroughMassachusettsExchangeFlag

 , to\_keep\_v1

 , count(\*) as nrecs

 from (

 select MemberEligibilityID

 , me.OrgID

 , me.SubmissionControlID

 , me.SubmissionYearMonth

 , InsuranceTypeCodeProduct

 , MedicalCoverage

 , PrimaryInsuranceIndicator

 , CoverageType

 , MarketCategoryCode

 , SpecialCoverage

 , cast(ProductIDNumber\_Linking\_ID as char(**20**)) as ProductID

 , prd.ProductLineOfBusinessModel

 , prd.Situs

 , ProductEnrollmentStartDate as StartDate

 , isnull(ProductEnrollmentEndDate,cast('99991231' as datetime)) as EndDate

 , LastActivityDate

 , HashCarrierSpecificUniqueMemberID as UniqueMemberID

 , MemberGenderCleaned as MemberGender

 , case when months\_between(Dates,ad.DT)/**12** >= **0** and months\_between(Dates,ad.DT)/**12** < **1** then '< 1'

 when months\_between(Dates,ad.DT)/**12** >= **1** and months\_between(Dates,ad.DT)/**12** < **10** then '1-9'

 when months\_between(Dates,ad.DT)/**12** >= **10** and months\_between(Dates,ad.DT)/**12** < **20** then '10-19'

 when months\_between(Dates,ad.DT)/**12** >= **20** and months\_between(Dates,ad.DT)/**12** < **27** then '20-26'

 when months\_between(Dates,ad.DT)/**12** >= **27** and months\_between(Dates,ad.DT)/**12** < **45** then '27-44'

 when months\_between(Dates,ad.DT)/**12** >= **45** and months\_between(Dates,ad.DT)/**12** < **65** then '45-64'

 when months\_between(Dates,ad.DT)/**12** >= **65** and months\_between(Dates,ad.DT)/**12** < **75** then '65-74'

 when months\_between(Dates,ad.DT)/**12** >= **75** and months\_between(Dates,ad.DT)/**12** < **85** then '75-84'

 when months\_between(Dates,ad.DT)/**12** >= **85** and months\_between(Dates,ad.DT)/**12** < **121** then '85+'

 else 'Unknown'

 end as AgeGroup

 , RiskAdjustmentCoveredBenefitPlan as RACP

 , case when MemberStateorProvince ='MA' then **1** else **0** end as MemberMA

 , case when MemberStateorProvince ='MA' then MemberCounty else 'Other' end as County

 , PurchasedThroughMassachusettsExchangeFlag

 , Dates

 , case when lead(me.MemberEligibilityID) over (partition by me.OrgID, HashCarrierSpecificUniqueMemberID, Dates

 order by case when MemberStateorProvince='MA' then **1** else **0** end

 , case when MedicalCoverage='1' then **1** else **0** end

 , case when PrimaryInsuranceIndicator in ('1','3','4') then **1** else **0** end

 , case when CoverageType in ('ASO', 'ASW') then **1**

 when CoverageType='UND' then **0**

 else -**1** end

 , LastActivityDate

 , me.MemberEligibilityID)

 is null then **1** else **0**

 end as to\_keep\_v1 /\* De-duplication hierarchy for Enrollment Trends \*/

 FROM APCD\_MemberEligibility as me

 inner join DateList

 on Dates between ProductEnrollmentStartDate and isnull(ProductEnrollmentEndDate,cast('99991231' as datetime))

 left join Product as prd

 on me.OrgID=prd.OrgID and me.ProductIDNumber\_Linking\_ID=prd.HashOrgIDProductIDNumber

 left join DATE\_TABLE as ad /\* Join to a date table to process only valid Member Date of Birth values \*/

 on me.MemberDateofBirthYearMonth||'01' = ad.DT\_YYYYMMDD

 where me.SubmissionYearMonth=&SYM.

 and me.OrgID in &OrgIdList.

 ) a

 where to\_keep\_v1=**1** /\* Save out only de-duplicated records \*/

 group by OrgID

 , SubmissionControlID

 , SubmissionYearMonth

 , Dates

 , InsuranceTypeCodeProduct

 , MedicalCoverage

 , PrimaryInsuranceIndicator

 , CoverageType

 , MarketCategoryCode

 , SpecialCoverage

 , ProductID

 , ProductLineOfBusinessModel

 , Situs

 , RACP

 , MemberMA

 , County

 , MemberGender

 , AgeGroup

 , PurchasedThroughMassachusettsExchangeFlag

 , to\_keep\_v1

 order by **1**,**2**,**3**,**4**

 );

 disconnect from odbc;

 **quit**;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

End: Netezza summary data extract

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\* Apply hard-coded groupings

 Macro Input: dataEdit(name of input dataset, name of output dataset)

 Macro Output: a temporary SAS dataset

;

**proc** **format**;

 value OrgName

 **290**, **7655**, **10353**, **10441**, **10442**, **10647**, **10929**, **11745** = 'Aetna'

 **10632** = 'Anthem'

 **291** = 'BCBS'

 **3505** = 'BMC'

 **10920** = 'CeltiCare'

 **293**, **295**, **7422**, **10447**, **11474**, **11499**, **11726**, **11215** = 'CIGNA'

 **296**, **8026** = 'Fallon'

 **300, 13074** = 'HPHC'

 **302** = 'HPI'

 **301** = 'HNE'

 **3156** = 'MassHealth'

 **12226** = 'Minuteman'

 **3735** = 'NHP'

 **4962** = 'NetworkHealth'

 **9913** = 'SWH'

 **8647** = 'Tufts'

 **312**, **313**, **7789**, **10444**, **10925**, **10926**, **10932**, **10933**, **10934**, **10935, 12814** = 'United'

 other = 'Other'

 ;

 value $MktSect

 'IND', 'GCV', 'ISCO' = 'Individual'

 'GS1', 'GS2', 'GS3' = 'Small Group (1-25)'

 'GS4' = 'Small Group (26-50)'

 'GLG1' = 'Mid-Size Group (51-100)'

 'GLG2', 'GLG3' = 'Large Group (101-500)'

 'GLG4' = 'Jumbo-Size Group (501+)'

 'GSA' = 'Qualified Association'

 other = 'Other'

 ;

 value $CovType

 'UND' = 'Fully-Insured'

 'ASO', 'ASW' = 'Self-Insured'

 other = 'Other'

 ;

 value $ProdType

 'MC', 'MO' = 'Medicaid'

 'CC' = 'Commonwealth Care'

 'CE' = 'Commonwealth Choice'

 'HM' = 'HMO'

 '12' = 'PPO'

 '13' = 'POS'

 '14' = 'EPO'

 '15' = 'Indemnity'

 '16', 'HN', '20' = 'Medicare'

 'SC' = 'Senior Care Options'

 other = 'Other'

 ;

 /\* Group NetworkHealth (OrgID 4962) under Tufts and HPI (OrgID 302) under HPHC

 for Enrollment Trends reporting purposes \*/

 value OrgNameReport

 **290**, **7655**, **10353**, **10441**, **10442**, **10647**, **10929**, **11745** = 'Aetna'

 **10632** = 'Anthem'

 **291** = 'BCBS'

 **3505** = 'BMC'

 **10920** = 'CeltiCare'

 **293**, **295**, **7422**, **10447**, **11474**, **11499**, **11726**, **11215** = 'CIGNA'

 **296**, **8026** = 'Fallon'

 **300**, **302, 13074** = 'HPHC'

 **301** = 'HNE'

 **3156** = 'MassHealth'

 **12226** = 'Minuteman'

 **3735** = 'NHP'

 **9913** = 'SWH'

 **4962**, **8647** = 'Tufts'

 **312**, **313**, **7789**, **10444**, **10925**, **10926**, **10932**, **10933**, **10934**, **10935, 12814** = 'United'

 other = 'Other'

 ;

 **run**;

**%macro** dataEdit(dataInfile, dataOutfile /\*, dataName\*/);

PROC SQL;

 CREATE TABLE &dataOutfile. AS

 SELECT \*

 , CASE WHEN ORGID=**300** AND INSURANCETYPECODEPRODUCT='15' THEN '2'

 when ORGID in (**290**, **7655**, **10353**, **10441**, **10442**, **10647**, **10929**, **11745**) and PRIMARYINSURANCEINDICATOR='3' then '2'

 ELSE PRIMARYINSURANCEINDICATOR

 END AS PRIMARYCOVERAGE

 , put(MarketCategoryCode, $MktSect.) as GROUPSIZE

 /\* - Finding: Fallon (OrgID=8026) incorrectly classified its self-insured population as

 InsuranceTypeCodeProduct='09'. As a result, cannot break out its self-

 insured population by product type. All other members are fully-insured.

 - Resolution: Need to hardcode MARKET='Self-Insured' for InsuranceTypeCodeProduct='09'

 and MARKET='Fully-Insured' for all other members for ET reporting. \*/

 , CASE WHEN ORGID=**8026** THEN

 CASE WHEN InsuranceTypeCodeProduct='09' THEN 'Self-Insured'

 ELSE 'Fully-Insured' END

 /\* Finding: United (OrgID=7789) is a student health plan, which is classified as

 CoverageType='STN'.

 Resolutions: Need to hardcode MARKET='Fully-Insured' for ET reporting. \*/

 WHEN ORGID=**7789** THEN

 CASE WHEN CoverageType in ('STN', 'UND') THEN 'Fully-Insured'

 WHEN CoverageType in ('ASO', 'ASW') THEN 'Self-Insured'

 ELSE 'Other' END

 ELSE put(CoverageType, $CovType.) end as MARKET

 /\* - Finding: Aetna (OrgID=10929) is supposed to be all Medicare members,

 but Aetna submits their Medicare members under various Product Type

 classifications ('12', 'HM', 'MD').

 - Resolution: Need to hardcode PLANTYPE='Medicare' for ET reporting. \*/

 , CASE WHEN ORGID=**10929** THEN 'Medicare'

 /\*HNE incorrectly classified members as InsuranceTypeCodeProduct=MP. Should be HM\*/

 WHEN ORGID=**12226** THEN

 CASE WHEN InsuranceTypeCodeProduct='MP' THEN 'HMO'

 else put(InsuranceTypeCodeProduct, $ProdType.) END

 ELSE CASE WHEN SpecialCoverage='CC' THEN 'Commonwealth Care'

 ELSE put(InsuranceTypeCodeProduct, $ProdType.) END

 END AS PLANTYPE

 , put(OrgId, OrgName.) as CompanyName

 , put(OrgID, OrgNameReport.) as CompanyName\_Report

 FROM &dataInfile.;

 QUIT;

\* Double check to make sure the County field is a true Massachusetts-based county;

 PROC SQL;

 CREATE TABLE &dataOutfile. AS

 SELECT \*

 , CASE WHEN UPCASE(County) in ('BARNSTABLE', 'BERKSHIRE', 'BRISTOL', 'DUKES', 'ESSEX',

 'FRANKLIN', 'HAMPDEN', 'HAMPSHIRE', 'MIDDLESEX', 'NANTUCKET', 'NORFOLK',

 'PLYMOUTH', 'SUFFOLK', 'WORCESTER', '') THEN County

 ELSE 'Other'

 END AS MEMBERCOUNTY

 FROM &dataOutfile.

 QUIT;

**%mend** dataEdit;

%***dataEdit***(out.ME\_ET\_&SYM., out.report\_&SYM.);

/\* Export data into txt-file to pivot in Power Pivot \*/

**proc** **sql** noprint;

 select '"'||trim(name)||'"'

 into :vars

 separated by "'|'"

 from dictionary.columns

 where upcase(libname)=upcase("out") and

 upcase(memname)=upcase("Report\_&SYM.")

 ;

 **quit**;

/\* PROCESSING TIME:

 \*/

filename outfile "&DataDir.\ET\_&SYM..txt";

**data** \_null\_;

 file outfile delimiter='|' DSD DROPOVER lrecl=**32767**;

 if \_n\_ = **1** then do;

 put &vars.;

 end;

 set out.Report\_&SYM. (obs=max);

 put (\_all\_) (+**0**) ;

 **run**;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

End: Export summary data extract

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

1. ConnectiCare was removed from reporting due to low Massachusetts enrollment. [↑](#footnote-ref-1)
2. Include ME028=(3, 4) if the payer stores primary insurance as such in its system. [↑](#footnote-ref-2)
3. This selection hierarchy is to ensure that self-insured plans are given preference over fully-insured plans for members with concurrent enrollment in both self-insured and fully-insured coverage. [↑](#footnote-ref-3)
4. CHIA is providing this methodology and programming code as a convenience. It has been prepared for informational purposes only and is based on information believed to be reliable.  The methodology and programming code are subject to change without notice. CHIA does not provide any guarantee or opinion on its accuracy.  CHIA disclaims any liability for the improper or incorrect use of the information contained herein. [↑](#footnote-ref-4)