02 November 2021

MHS GENESIS Admission File
BDE 2.4 for the
MHS Data Repository (MDR)
(Version 1.03.00)

Current Specification

Revision History

Version	Date	Originator	Para/Tbl/Fig	Description of Change
1.00.00	03/09/19	Douglas Juckett Mitzi Miller	• Initial Document	• Initial Document

1.01.00	06/05/20	Douglas Juckett
		Mitzi Miller
			• Table 1
			• Table 2
				• Split Table 1 into Table 1a with data file merges and Table 1b with format file sources
				• Add AHRQ Indicator variables
				  • ASTDIAB
				  • APAPPD
				  • ALTDIAB
				  • ACOPD
				  • AHYPTN
				  • ACHF
				  • ALBW
				  • ADHYD
				  • ABACPN
				  • AUTI
				  • AAWP
				  • AUNCDIAB
				  • AASTH
				  • AAMPDIAB
				  • AOVALL
				  • AACUTE
				  • ACHRON
				  • PASTH
				  • PSTDIAB
				  • PGASTRO
				  • PAPPD
				  • PUTI
				  • POVALL
				  • PCHRON
				  • PACUTE
				  • PADCDOVL
				  • PADCDCHN
				  • PADCDACT
				  • AHRQPVADM
				• Add REASON_FOR_VISIT
				• Add SERVICE_LINE

1.02.00	11/24/20	Douglas Juckett
		Mitzi Miller
			• Table 1b
			
			• Table 2
				• Modify MSDRGRWP calculation to indicate that use CY starting in CY20
				• Add MDC & any other reference files to this table
				• Add SERVICE_LINE_1 – SERVICE_LINE_8
				• Change format for SERVICE_LINE to CHAR(5)
				• Add Provider Name variables
				  • PROV_NAME_ATT
				  • PROV_NAME_ADM

Version	Date	Originator	Para/Tbl/Fig	Description of Change
				• PROV_NAME2-PROV_NAME6 • PROV_NAME_PRIM
1.03.00	11/02/21	Douglas Juckett Mitzi Miller	• Table 1b • Table 2 • Appendix B • Appendix G	• Modify Service Line format file location • Modify ADM_INFR_FLAG definition • Update MSDRG input file specifications • Update flow chart

MDR GENESIS ADMISSION FILE

I. SOURCE

The source system is the Cerner Millennium. All records are based on records sent from the WH_CLN_ENCOUNTER file. In order to increase the utility of this file, variables from files in the following Cerner subject areas have been added: Encounters, Appointment, Proc_Diag, Person, Personnel, Location, and various reference files. For the same reason, variables from the DMHRSi-HR, Master Person Index (MPI), DEERS longitudinal enrollment, DMIS ID Index and Omni-CAD files have been added.

II. TRANSMISSION (FORMAT AND FREQUENCY)

Weekly.

III. ORGANIZATION AND BATCHING

Source data: The first step in MDR processing is to batch records received from MHS GENESIS. Raw data batches are stored in mdr/raw/genesis according to routine MDR operating procedures.

Output products: SAS dataset containing all admission records where the ADM_DT_TM is in a fiscal year. The Admission files are stored at /mdr/pub/genesis/admission/fyxx.sas7bdat.

IV. RECEIVING FILTERS

Records are included/excluded within the Encounters table as follows:

- Health_System_ID = 18635 or 91970 are included
- PATIENT_TYPE_REF are included if they are in the following:
    - 309308 - Inpatient

V. UPDATE PROCESS

All records were provided with the initial batch of data. Thereafter, new and updated records are sent each week. New records are added to the existing dataset. Updated records replace the original record in the existing dataset based on the Encounter_NK, which is the primary key for the Encounter Table. If there are multiple records with the same Encounter_NK, the updated record with the most recent UPDT_DT_TM is kept. Similar processes are applied to the other raw data sets. FY is determined for both the Admission and Discharge date and are split into separate files based on Discharge date, if available, and Admission date, if not.

Once the raw data has been updated, the processor combines them as described here and assigns many other internally-derived variables as described in Table 2.

VI. FIELD TRANSFORMATIONS AND DELETIONS FOR MDR CORE DATABASE

This section of this functional specification describes the data merges that are necessary to append fields in the MDR GENESIS Admission file. Table 1a describes the reference files that are used in processing. Table 1b describes the format/reference files used in processing. Table 2 lists in detail all the fields added from these merges as well as any additional transformation rules.

TABLE 1a: Data File Merges for MDR GENESIS Admission File

Merge	Matching	Matching Methodology, Limitations and Test Record Removal
ENCOUNTER (enco_cencounter) to LOCATION (mdr/pub/ genesis/location) Removing test records and adding fields related to the location of the encounter such as treatment DMIS ID, MEPRS code, etc.	ENCOUNTER Current_Loc LOCATION Location_SK	Keep records where • test_location_ind=0 and • datepart(adm_dt_tm) >= gen_begin_dt
ENCOUNTER (enco_cencounter) to PERSON (mdr/pub/ genesis/person) Removing test patient IDs and adding fields related to the patient such as EDIPN, gender, date of birth, etc.	Person_SK	Keep records where • test_record_ind = 0.
PERSON (mdr/pub/ genesis/person) to RAW_USER_DEF_HIST (enco_rudhist) Adding Patient Category associated with the GENESIS Person file.. ENCOUNTER (enco_cencounter) to RAW_USER_DEF_HIST (enco_rudhist) Adding Patient Category associated with the GENESIS Encounter file.	PERSON Person_SK USER_DEF Parent_Entity_SK ENCOUNTER Encounter_SK USER_DEF Parent_Entity_SK	PERSON_SK = Parent_Entity_SK when Parent_Entity_Name = PERSON & User_Defined_Type_Ref = 114540103 Encounter_SK = Parent_Entity_SK when Parent_Entity_Name = ENCOUNTER & User_Defined_Type_Ref = 109901051 or 109901057
ENCOUNTER (enco_cencounter) to ENC_HISTORY (enco_cehistory) 1st Merge: Adding Original Extract Date for each Encounter 2nd Merge: Adding all locations that occur during the inpatient stay	Encounter_NK	1st Merge: Keep records where • Extract_DT_TM is the earliest value for each Encounter_NK 2nd Merge: Keep records where • Encounter Type of latest record for an Encounter_NK is Inpatient and • Encounter Type on other records for the same Encounter_NK is not Preadmit, Prereg, Emergency, Observation, Outpatient, Outpatient in a Bed, Outpatient Hold, Outpatient Day Surgery, Referral Tracking, PreRecurring, Clinic or Dental. • Transact_DT_TM is between the dates of the inpatient admission and discharge. If no discharge yet then is after the inpatient admission date.

Merge	Matching	Matching Methodology, Limitations and Test Record Removal
ENCOUNTER (enco_cencounter) to ENC_GROUPER (enco_cegroupings) Adding Cerner calculated MS-DRG & MSMDC data.	Encounter_NK	1st Merge: Keep records where • UPDT_DT_TM is the last value for each Encounter_NK
ENCOUNTER (enco_cencounter) to ENC_PRSNL_RELTN (enco_cenpreltn) Identifying providers associated with the encounter and adding provider ID and role for selected ones. Plus identifying and adding the ID and role of up to 4 coders.	Encounter_Key	Delete records where Personnel_Type_Ref (format = $personnel_type.), called Personnel_Type, contains any of the following words: HIM, Database, View Only, Clerk, Auditor, Inbox, Planner, Other, Student, Practice Management User, or Administrator. Put all records on one line per Encounter_Key. Keep only the 1st instance of records where Personnel_Type = Attending Physician, Admitting Physician and/or Referring Physician and up to the next 5 Encounter_Prsnl values where records with the Personnel_Type containing the word Physician are given priority. Keep up to the 1st 4 unique Encounter_Prsnl values for each Encounter_Key where Personnel_Type contains 'Coder' or 'Coding.'
ENCOUNTER (enco_cencounter) to PERSONNEL (mdr/pub/ genesis/personnel) Removing test personnel from the encounter and adding fields such as provider EDIPN, NPI and specialty for al l providers included on the Encounter record.	<u>ENCOUNTER</u> PROVID2-PROVID6 PROVID_ATT PROVID_ADM PROVID_REF CODERID1-CODERID4 <u>APPOINTMENT</u> PROVID1 <u>PERSONNEL</u> PRSNL_SK	Keep records where • Test_Record_Ind = 0 and • Prsnl_Key > 0.
ENCOUNTER (enco_cencounter) to APPOINTMENT (mdr/pub/genesis/ appointment) Adding appointment-related fields such as Appointment Type, Appointment Status, Appointment Provider, etc.	Encounter_Key	Remove records where Encounter_Key is • <=0 (invalid Encounter_Key or appointments that did not generate an encounter). Exclude records where there are more than one Appointment record with the same Encounter_Key. Add back those records that match based on the hierarchy below. If have

Merge	Matching	Matching Methodology, Limitations and Test Record Removal
		more than one record per Encounter_Key as a result of the hierarchy then keep the one with the highest match type value. <u>Match Hierarchy (Appointment to Encounter):</u> 1. Non-duplicated Encounter Key 2. Duplicated Encounter Key, Person, Date, Location, Provider 3. Duplicated Encounter Key, Person, Date, Location 4. Duplicated Encounter Key, Person, Date
ENCOUNTER (enco_cencounter) to DIAGNOSIS (prdi_cdiagnosis) Adding a select number of diagnosis fields associated with the encounter plus an indicator of whether the diagnoses for that encounter have been through CCE.	Encounter_NK	Keep records where • [contributor_system_ref = 110586353 (Inbound Coding from CCE) or • (contributor_system_ref = 469 (PowerChart) and datepart(enc_dt_tm) > gen_begin_dt] or • ENCOUNTER_KEY > 0 or • PRIORITY_SEQ < 99. Remove records where Encounter_NK is invalid, including 018635, 0~18635 and 0~0.
ENCOUNTER (enco_cencounter) to PROCEDURE (prdi_cprocedure) Adding procedure codes associated with the encounter, including ICD10PROC codes, which have gone through CCE.	Encounter_NK	The following records are included after the merge: • PRIORITY_SEQ >= 99 (ICD10 Procedure), • PRIORITY_SEQ = . (no Procedure)
ENCOUNTER (enco_cencounter) to DMHRSi Adding provider-related fields such as assigned MTF, assigned MEPRS, etc.	PROV_EDIPN# PROV_NPI#	See Appendix A. There are separate merges for every provider ID variable, excluding the 4 Coder IDs.
ENCOUNTER (enco_cencounter) to Master Person Index (MPI) Adding patient-related fields from DEERS such as EDIPN, PARC, SPONSSN, etc.	Patient EDIPN and/or SSN, GENDER_R, DOB_R, Last_Name, First_Name	See Appendix A.
ENCOUNTER (enco_cencounter) to DEERS (LVM) Adding patient-related fields from DEERS such as ACV, gender, race, date of birth, etc.	Patient EDIPN Sponsor SSN	See Appendix A.

Merge	Matching	Matching Methodology, Limitations and Test Record Removal
ENCOUNTER (enco_cencounter) to DMIS ID Index Table Adding DMIS ID-related fields such as branch of service, T17 region, etc.	FY from ENC_DT_TM MTF ENR_MTF	See Appendix A.
ENCOUNTER (enco_cencounter) to Omni-CAD Adding patient-related fields such as catchment, PRISM, beneficiary T3 region,, etc.	PATZIP SPONSVC	See Appendix A.
ENCOUNTER (enco_cencounter) to MS-DRG Grouper Adding patient-related fields such as DRG & MDC, etc. using the TRICARE MS-DRG grouper.	FYDISP FYADM ENCOUNTER_NK	Keep records where • DISCHARGE_DT_TM is not missing
ENCOUNTER (enco_cencounter) to CLINICAL_EVENT (clev_ccevent) Adding birth weight data from Clinical Events file.	ENCOUNTER_NK	Keep records where • UPDT_DT_TM is the last value for each Encounter_NK • EVENT_CODE_REF is '712070', '29335541' or '17022176'
PREGNANCY INSTANCE (mate_cpinst) to PREGNANCY CHILD (mate_cpchild) to ENCOUNTER (enco_cencounter) to ENCOUNTER PERSON RELATION (enco_ceereltn) Adding birth weight data from Clinical Events file.	<u>Pregnancy Child</u> PREGNANCY_INST_SK <u>Encounter & Encounter Person Relation</u> ENCOUNTER_NK RELATED_ENCOUNTER_NK	Keep records where • SRC_ACTIVE_IND = 1 • HISTORICAL_IND = 1 • PREG_END_TM_VLD_FLG = 1 • Latest UPDT_DT_TM Keep records where • Latest UPDT_DT_TM

TABLE 1b: Format File Merges for MDR GENESIS Admission File

Merge	Matching	Source
GENESIS BDE Format File Get formatted values for many different GENESIS variables.	Many	/mdr/int/genesis/basic/&ddate./gns_formats.sas7bcat &ddate. = BDE Extract Date (dYYMMDD)
RWP Format File – Real Records	FYDISP FYADM	Thru FY19/CY19 : /mdr_ref/sidr.msdrgwts.fyYY CY20 +: /mdr_ref/sidr.msdrgwts.cyYY

Merge	Matching	Source
Adding RWP-related using the MS-DRG RWP format file.	MSDRG ADM_INFR_FLAG = 0	YY = Fiscal Year based on Dispositioning FY, if available, or Admitting FY, if not
RWP Format File – Inferred Records Adding RWP-related using the MS-DRG RWP format file.	FYDISP FYADM MSDRG ADM_INFR_FLAG = 1	/mdr_ref/sidr.inferred.fyYY YY = Fiscal Year based on Dispositioning FY, if available, or Admitting FY, if not
Medical/Surgical Format File Adding medical/surgical indicator.	MSDRG	/mdr_ref/sidr.msdrgsurg
HIPAA Taxonomy-based Skill Type Format File Adding Skill Type based on HIPAA Taxonomy	PROV_HIPAA_### ### = Valid value for each provider	/mdr/ref/caper.hskilltype.fyYY.txt YY = Fiscal Year based on Dispositioning FY, if available, or Admitting FY, if not
Service Line Format File Adding Service Line.	MEPRS_CD MEPR3	/mdr_ref/ slfmt.fmt
AHRQ Indicators Format Files Adding AHRQ indicators.	DX1 – DX20 PROC_1 – PROC_20	/mdr_ref/ahrq.icd10/pdfmts.sas /mdr_ref/ahrq.icd10/pqfmts.sas

Upon matching to the MDR GENESIS Person file, MDR GENESIS Personnel file, and MDR GENESIS Location file, records that meet the following criteria that define them as test records are deleted:

- Records where the test_person_ind=1 in the MDR GENESIS Person File.
- Records where any of the test_personnel_ind fields = 1 in the MDR GENESIS Personnel File.
- Records where the test_location_ind=1 in the MDR GENESIS Location File.
- Records where the admission or discharge date is before the go live date for the location for the MTF.

VII. FILE LAYOUT

The MDR GENESIS Admission file is stored in a SAS data set. Table 2 provides the file layout and processing rules.

TABLE 2: Fields in the MDR GENESIS Admission

Field	Format	SAS Name	Source Element	Transformation
CERNER ENCOUNTER (enco_cencounter):				
Admission Source	CHAR(43)	ADMIT_SOURCE	ADMIT_SOURCE_REF	Apply format $admit_source (code set = 2)
Admission Type	CHAR(32)	ADMIT_TYPE	ADMIT_TYPE_REF	Apply format $admit_type (code set = 3)
CCE Admission Status Flag	N(8)	CCESTAT	CODING_COMPLETED_TM_VLD_FLAG	No transformation.
Admission Location (Numeric Code)	CHAR(40)	CURRENT_LOC	CURRENT_LOC	No transformation.
Discharge Disposition Code	CHAR(60)	DISPCODE	DISCHARGE_DISPOSITION_REF	Apply format $discharge_disposition (code set = 19)
Encounter Class	CHAR(33)	ENCOUNTER_CLASS	ENCOUNTER_TYPE_CLASS_REF	Apply format $encounter_type_class. (code set = 69)
Encounter Key (Primary)	CHAR(100)	ENCOUNTER_NK	ENCOUNTER_NK	No transformation.
Encounter Key (Secondary)	CHAR(100)	ENCOUNTER_SK	ENCOUNTER_SK	No transformation.
Encounter PI-EDW Key	N(8)	ENCOUNTER_KEY	ENCOUNTER_KEY	No transformation.
Encounter Status	CHAR(25)	ENCOUNTER_STATUS	ENCOUNTER_STATUS_REF	Apply format $encounter_status. (code set = 261)
Encounter Type	CHAR(26)	ENCOUNTER_TYPE	PATIENT_TYPE_REF	Apply format $patient_type. (code set = 71)
Financial Classification (Payment Source)	CHAR(37)	FINANCIAL_CLASS	FINANCIAL_CLASS_REF	Apply format $financial_class. (code set = 354)
Financial Information Number (FIN)	CHAR(40)	FIN	FORMATTED_FINANCIAL_NBR	No transformation.
GENESIS Extract Date	Date/Time	EXTRDATE_UTC	EXTRACT_DT_TM	Apply format e8601dt.
GENESIS Status	N(8)	GEN_STATUS	TOTAL_UPDATES	No transformation.
Health Facts MS-DRG (Cerner)	CHAR(3)	HF_CALC_DRG	HF_CALC_DRG	No transformation.
Health Facts MDC (Cerner)	CHAR(2)	HF_CALC_MDC	HF_CALC_MDC	No transformation.
Health System ID	N(8)	HEALTH_SYSTEM_ID	HEALTH_SYSTEM_ID	No transformation.
Medical Service	CHAR(40)	MEDICAL_SVC	MEDICAL_SERVICE_REF	Apply format $medical_service. (code set = 34)
MRN (Encounter) (NOT RETAINED)	CHAR(40)	FORMATTED_MRN	FORMATTED_MRN	No transformation.
Person File Primary Key	CHAR(100)	PERSON_SK	PERSON_SK	No transformation.
Reason for Visit (Free Text)	CHAR(255)	REASON_FOR_VISIT	REASON_FOR_VISIT_TXT	No transformation.
MDR LOCATION (location)				
Admission Location (Composite Description)	CHAR(45)	ADMISSION_LOC_COMPOSITE	UNIT_DISPLAY	No transformation.
Clinic State of Record	CHAR(50)	CLINSTAT_R	STATE	No transformation.
Clinic Zip Code of Record	CHAR(25)	CLINZIP_R	POSTAL_CODE	No transformation.

Field	Format	SAS Name	Source Element	Transformation
Admission Location (General)	CHAR(50)	ADMISSION_LOC_GENERAL	UNIT_NAME	No transformation.
GENESIS Data Begin Date	N(8)	GEN_BEGIN_DT	GEN_BEGIN_DT	Format date as MMDDYY10.
MEPRS Reporting Status of MTF	N(8)	NOMEPRS_FLAG	NOMEPRS_FLAG	No transformation.
Military Treatment Facility (MTF)	CHAR(4)	MTF	MTF	No transformation.
Test Location Flag	N(8)	TEST_LOCATION_FLAG	TEST_LOCATION_FLAG	No transformation.
Treatment MEPRS Code	CHAR(4)	MEPRS_CD	MEPRS_CD	No transformation.
Treatment Parent MTF (DMIS ID Index File) (NOT RETAINED)	CHAR(4)	MTF_PARENT_G	MTF_PARENT	No transformation.
Treatment Parent MTF (MHS GENESIS)	CHAR(4)	MTF_PARENT_REC	MTF_PARENT_REC	No transformation.
MDR PERSON (person)				
MRN (Person) (NOT RETAINED)	CHAR(40)	MRNPERSON	MRN	No transformation.
Patient Date & Time of Birth of Record (NOT RETAINED)	Date/Time	DOB_R	DOB_R	No transformation.
Patient First Name of Record (NOT RETAINED)	CHAR(20)	FIRST_NAME	FIRST_NAME	No transformation.
Patient Gender of Record	CHAR(10)	GENDER_R	GENDER_R	No transformation.
Patient Last Name of Record (NOT RETAINED)	CHAR(26)	LAST_NAME	LAST_NAME	No transformation.
Patient Social Security Number of Record (NOT RETAINED)	CHAR(9)	SSN	SSN	No transformation.
Test Person Flag	N(8)	TEST_PERSON_FLAG	TEST_RECORD_IND	No transformation.
Unique Person Identifier of Record (EDIPN) (NOT RETAINED)	CHAR(10)	EDIPN	EDIPN	No transformation.
CERNER RAW USER DEFINED HISTORY (enco_rudhist)				
BENCAT of Record (MHS GENESIS)	CHAR(40)	BENCAT_E	VALUE_REF	Apply format $bencat (code set = 100070)
Patient Category (MHS GENESIS ENCOUNTER)	CHAR(40)	PATCAT_E	VALUE_REF	Apply format $patcat (code set = 100075)
Patient Category (MHS GENESIS PERSON)	CHAR(40)	PATCAT_P	VALUE_REF	Apply format $patcat (code set = 100075)
CERNER ENCOUNTER HISTORY (enco_cehistory)				
MDR Original Encounter Extract Date	CHAR(8)	EXTRDATE_O	EXTRACT_DT_TM	put(datepart(extract_dt_tm),yymmddn8.)

Field	Format	SAS Name	Source Element	Transformation
Admission Location (Enc History) (NOT RETAINED)	CHAR(40)	LOC_NURSE_UNIT_REF	LOC_NURSE_UNIT_REF	No transformation.
Encounter Type (Enc History) (NOT RETAINED)	CHAR(26)	ENCOUNTER_TYPE_EH	ENCNTR_TYPE_REF	Apply format $patient_type. (code set = 71)
Medical Service (Enc History) (NOT RETAINED)	CHAR(40)	MEDICAL_SERVICE_EH	MEDICAL_SERVICE_REF	Apply format $medical_service. (code set = 34)
CERNER DRG/MDC GROUPER (enco_cegroupings)				
MS-DRG (Cerner)	CHAR(36)	GEN_MSDRG	GROUPER_NOMEN	put(grouper_nomen,$nomen) Prior to January 2019, the GEN_MSDRG is actually the Medicare MS-DRG grouper not the TRICARE MS-DRG one. There is no plan to rerun the prior data using the TRICARE MS-DRG grouper.
MS-DRG Average Length of Stay (Cerner)	N(8)	AVERAGE_LOS	AVERAGE_LOS	No transformation.
MS-DRG Comorbidity (Cerner)	CHAR(40)	COMORBIDITY_REF	COMORBIDITY_REF	No transformation.
MS-DRG Contributor System (Cerner)	CHAR(60)	CONTRIBUTOR_SYSTEM_DRG	CONTRIBUTOR_SYSTEM_REF	put(contributor_system_ref,$contributor_system.)
MS-DRG High Trim Value (Cerner)	N(8)	HIGH_TRIM_VALUE	HIGH_TRIM_VALUE	No transformation.
MS-DRG Low Trim Value (Cerner)	N(8)	LOW_TRIM_VALUE	LOW_TRIM_VALUE	No transformation.
MS-DRG Major Complicating Condition (Cerner)	N(8)	MCC	MCC	No transformation.
MS-DRG Major Complicating Condition Text (Cerner)	CHAR(255)	MCC_TEXT	MCC_TEXT	No transformation.
MS-DRG Patient Status (Cerner)	CHAR(40)	PATIENT_STATUS_REF	PATIENT_STATUS_REF	No transformation.
MS-DRG Vocabulary (Cerner)	CHAR(60)	VOCABULARY_DRG	VOCABULARY_REF	Use Code Set = 400.
MS-MDC (Cerner)	CHAR(60)	GEN_MDC	MDC_REF	put(mdc_ref,$drg_mdc)
CERNER ENCOUNTER PERSONNEL RELATIONS (enco_cenpreltn)				
Coding Personnel 1 ID	CHAR(100)	CODER_ID1	ENCOUNTER_PRSNL	If formatted Personnel_Type_Ref (format = $personnel_type.), called Personnel_Type, contains 'Coder' or 'Coding' then CODERID1 = ENCOUNTER_PRSNL
Coding Personnel 1 Role	CHAR(40)	CODER_ROLE1	PERSONNEL_TYPE_REF	Personnel_type_ref where coderid1=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Coding Personnel 2 ID	CHAR(100)	CODER_ID2	ENCOUNTER_PRSNL	If formatted Personnel_Type_Ref (format = $personnel_type.), called Personnel_Type,

Field	Format	SAS Name	Source Element	Transformation
				contains 'Coder' or 'Coding' then CODERID2 = ENCOUNTER_PRSNL
Coding Personnel 2 Role	CHAR(40)	CODER_ROLE2	PERSONNEL_TYPE_REF	Personnel_type_ref where coderid2=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Coding Personnel 3 ID	CHAR(100)	CODER_ID3	ENCOUNTER_PRSNL	If formatted Personnel_Type_Ref (format = $personnel_type.), called Personnel_Type, contains 'Coder' or 'Coding' then CODERID3 = ENCOUNTER_PRSNL
Coding Personnel 3 Role	CHAR(40)	CODER_ROLE3	PERSONNEL_TYPE_REF	Personnel_type_ref where coderid3=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Coding Personnel 4 ID	CHAR(100)	CODER_ID4	ENCOUNTER_PRSNL	If formatted Personnel_Type_Ref (format = $personnel_type.), called Personnel_Type, contains 'Coder' or 'Coding' then CODERID4 = ENCOUNTER_PRSNL
Coding Personnel 4 Role	CHAR(40)	CODER_ROLE4	PERSONNEL_TYPE_REF	Personnel_type_ref where coderid4=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Provider ID For Admitting Physician (MHS GENESIS)	CHAR(100)	PROVID_ADM	ENCOUNTER_PRSNL	If formatted Personnel_Type_Ref (format = $personnel_type.), called Personnel_Type, is 'Admitting Physician' then PROVID_ADM = ENCOUNTER_PRSNL
Provider ID For Attending Physician (MHS GENESIS)	CHAR(100)	PROVID_ATT	ENCOUNTER_PRSNL	If formatted Personnel_Type_Ref (format = $personnel_type.), called Personnel_Type, is 'Attending Physician' then PROVID_ATT = ENCOUNTER_PRSNL
Provider ID For Provider 2 (MHS GENESIS)	CHAR(100)	PROVID2	ENCOUNTER_PRSNL	Sort by encounter_key, physician (descending) and then transpose to get all providers on one line with physicians 1st. Select the 1st Prov_# that does not equal the Attending, Admitting or Referring Provider ID.
Provider ID For Provider 3 (MHS GENESIS)	CHAR(100)	PROVID3	ENCOUNTER_PRSNL	Sort by encounter_key, physician (descending) and then transpose to get all providers on one line with physicians 1st. Select the 2nd Prov_# that does not equal the Attending, Admitting or Referring Provider ID.
Provider ID For Provider 4 (MHS GENESIS)	CHAR(100)	PROVID4	ENCOUNTER_PRSNL	Sort by encounter_key, physician (descending) and then transpose to get all providers on one line with physicians 1st. Select the 3rd Prov_# that does not equal the Attending, Admitting or Referring Provider ID.
Provider ID For Provider 5 (MHS GENESIS)	CHAR(100)	PROVID5	ENCOUNTER_PRSNL	Sort by encounter_key, physician (descending) and then transpose to get all providers on one line with physicians 1st. Select the 4th Prov_# that does not equal the Attending, Admitting or Referring Provider ID.
Provider ID For Provider 6 (MHS GENESIS)	CHAR(100)	PROVID6	ENCOUNTER_PRSNL	Sort by encounter_key, physician (descending) and then transpose to get all providers on one line with physicians 1st. Select the 5th Prov_# that does not equal the Attending, Admitting or Referring Provider ID.

Field	Format	SAS Name	Source Element	Transformation
Provider ID For Referring Physician (MHS GENESIS)	CHAR(100)	PROVID_REF	ENCOUNTER_PRSNL	If formatted Personnel_Type_Ref (format = $personnel_type.), called Personnel_Type, is 'Referring Physician' then PROVID_REF = ENCOUNTER_PRSNL
Provider Role For Additional Provider 1 (MHS GENESIS)	CHAR(40)	PROV_ROLE2	PERSONNEL_TYPE_REF	Personnel_type_ref where provid2=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Provider Role For Additional Provider 2 (MHS GENESIS)	CHAR(40)	PROV_ROLE3	PERSONNEL_TYPE_REF	Personnel_type_ref where provid3=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Provider Role For Additional Provider 3 (MHS GENESIS)	CHAR(40)	PROV_ROLE4	PERSONNEL_TYPE_REF	Personnel_type_ref where provid4=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Provider Role For Additional Provider 4 (MHS GENESIS)	CHAR(40)	PROV_ROLE5	PERSONNEL_TYPE_REF	Personnel_type_ref where provid5=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Provider Role For Additional Provider 5 (MHS GENESIS)	CHAR(40)	PROV_ROLE6	PERSONNEL_TYPE_REF	Personnel_type_ref where provid6=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Provider Role For Admitting Physician (MHS GENESIS)	CHAR(40)	PROV_ROLE_ADM	PERSONNEL_TYPE_REF	Personnel_type_ref where provid_adm=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Provider Role For Appointment Provider (MHS GENESIS)	CHAR(40)	PROV_ROLE1	PERSONNEL_TYPE_REF	If nonblank then value is 'Appointment Provider'
Provider Role For Attending Physician (MHS GENESIS)	CHAR(40)	PROV_ROLE_ATT	PERSONNEL_TYPE_REF	Personnel_type_ref where provid_att=encounter_prnsl. Apply format $personnel_type. (code set = 333)
Provider Role For Referring Physician (MHS GENESIS)	CHAR(40)	PROV_ROLE_REF	PERSONNEL_TYPE_REF	Personnel_type_ref where provid_ref=encounter_prnsl. Apply format $personnel_type. (code set = 333)
MDR PERSONNEL (personnel)				
Primary HIPAA Taxonomy For Additional Provider 1	CHAR(10)	PROV_HIPAA2	HIPAA1	No transformation.
Primary HIPAA Taxonomy For Additional Provider 2	CHAR(10)	PROV_HIPAA3	HIPAA1	No transformation.
Primary HIPAA Taxonomy For Additional Provider 3	CHAR(10)	PROV_HIPAA4	HIPAA1	No transformation.
Primary HIPAA Taxonomy For Additional Provider 4	CHAR(10)	PROV_HIPAA5	HIPAA1	No transformation.
Primary HIPAA Taxonomy For Additional Provider 5	CHAR(10)	PROV_HIPAA6	HIPAA1	No transformation.
Primary HIPAA Taxonomy For Admitting Physician	CHAR(10)	PROV_HIPAA_ADM	HIPAA1	No transformation.
Primary HIPAA Taxonomy For Attending Physician	CHAR(10)	PROV_HIPAA_ATT	HIPAA1	No transformation.
Primary HIPAA Taxonomy For Referring Physician	CHAR(10)	PROV_HIPAA_REF	HIPAA1	No transformation.

Field	Format	SAS Name	Source Element	Transformation
Provider EDIPN For Additional Provider 1	CHAR(10)	PROV_EDIPN2	PRSNL_EDIPN	No transformation.
Provider EDIPN For Additional Provider 2	CHAR(10)	PROV_EDIPN3	PRSNL_EDIPN	No transformation.
Provider EDIPN For Additional Provider 3	CHAR(10)	PROV_EDIPN4	PRSNL_EDIPN	No transformation.
Provider EDIPN For Additional Provider 4	CHAR(10)	PROV_EDIPN5	PRSNL_EDIPN	No transformation.
Provider EDIPN For Additional Provider 5	CHAR(10)	PROV_EDIPN6	PRSNL_EDIPN	No transformation.
Provider EDIPN For Admitting Physician	CHAR(10)	PROV_EDIPN_ADM	PRSNL_EDIPN	No transformation.
Provider EDIPN For Attending Physician	CHAR(10)	PROV_EDIPN_ATT	PRSNL_EDIPN	No transformation.
Provider EDIPN For Referring Physician	CHAR(10)	PROV_EDIPN_REF	PRSNL_EDIPN	No transformation.
Provider Name For Additional Provider 1	CHAR(200)	PROV_NAME2	PERSONNEL_FULL_NAME	No transformation.
Provider NAME For Additional Provider 2	CHAR(200)	PROV_NAME3	PERSONNEL_FULL_NAME	No transformation.
Provider NAME For Additional Provider 3	CHAR(200)	PROV_NAME4	PERSONNEL_FULL_NAME	No transformation.
Provider NAME For Additional Provider 4	CHAR(200)	PROV_NAME5	PERSONNEL_FULL_NAME	No transformation.
Provider NAME For Additional Provider 5	CHAR(200)	PROV_NAME6	PERSONNEL_FULL_NAME	No transformation.
Provider NAME For Admitting Physician	CHAR(200)	PROV_NAME_ADM	PERSONNEL_FULL_NAME	No transformation.
Provider NAME For Attending Physician	CHAR(200)	PROV_NAME_ATT	PERSONNEL_FULL_NAME	No transformation.
Provider NAME For Referring Physician	CHAR(200)	PROV_NAME_REF	PERSONNEL_FULL_NAME	No transformation.
Provider NPI For Additional Provider 1	CHAR(10)	PROV_NPI2	NPI	No transformation.
Provider NPI For Additional Provider 2	CHAR(10)	PROV_NPI3	NPI	No transformation.
Provider NPI For Additional Provider 3	CHAR(10)	PROV_NPI4	NPI	No transformation.
Provider NPI For Additional Provider 4	CHAR(10)	PROV_NPI5	NPI	No transformation.
Provider NPI For Additional Provider 5	CHAR(10)	PROV_NPI6	NPI	No transformation.
Provider NPI For Admitting Physician	CHAR(10)	PROV_NPI_ADM	NPI	No transformation.
Provider NPI For Attending Physician	CHAR(10)	PROV_NPI_ATT	NPI	No transformation.
Provider NPI For Referring Physician	CHAR(10)	PROV_NPI_REF	NPI	No transformation.
Test Personnel Flag - Additional Provider 1	CHAR(1)	TEST_RECORD_P2	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Additional Provider 2	CHAR(1)	TEST_RECORD_P3	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Additional Provider 3	CHAR(1)	TEST_RECORD_P4	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Additional Provider 4	CHAR(1)	TEST_RECORD_P5	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Additional Provider 5	CHAR(1)	TEST_RECORD_P6	TEST_RECORD_IND	No transformation.

Field	Format	SAS Name	Source Element	Transformation
Test Personnel Flag - Admitting Physician	CHAR(1)	TEST_RECORD_ADM	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Appointment Provider	CHAR(1)	TEST_RECORD_P1	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Attending Physician	CHAR(1)	TEST_RECORD_ATT	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Coder 1	CHAR(1)	TEST_RECORD_C1	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Coder 2	CHAR(1)	TEST_RECORD_C2	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Coder 3	CHAR(1)	TEST_RECORD_C3	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Coder 4	CHAR(1)	TEST_RECORD_C4	TEST_RECORD_IND	No transformation.
Test Personnel Flag - Referring Physician	CHAR(1)	TEST_RECORD_REF	TEST_RECORD_IND	No transformation.
MDR APPOINTMENT (appointment)				
Appointment Date & Time	N(8)	APPT_DT_TM	APPT_DT_TM	No transformation. Format DATETIME19.
Appointment File Primary Key	CHAR(100)	SCH_APPOINTMENT_SK	SCH_APPOINTMENT_SK	No transformation.
Appointment Status Type	CHAR(20)	APPT_STAT	APPT_STAT	No transformation.
Appointment Type	CHAR(16)	APPT_TYPE	APPT_TYPE	No transformation.
Assigned Appointment Duration	CHAR(3)	ASSGNDUR	DURATION_R	No transformation.
Legacy Appointment Status Type	CHAR(2)	APPT_STAT_LEGACY	APPT_STAT_LEGACY	No transformation.
Legacy Appointment Type	CHAR(6)	APPT_TYPE_LEGACY	APPT_TYPE_LEGACY	No transformation.
Primary HIPAA Taxonomy For Appointment Provider	CHAR(10)	PROV_HIPAA1	HIPAA1	No transformation.
Provider EDIPN For Appointment Provider	CHAR(10)	PROV_EDIPN1	PROV_EDIPN1	No transformation.
Provider ID For Appointment Provider (MHS GENESIS)	CHAR(100)	PROVID1	PROVID1	No transformation.
Provider NPI For Appointment Provider	CHAR(10)	PROV_NPI1	PROV_NPI1	No transformation.
CERNER DIAGNOSIS (prdi_diagnosis)				
CCE Diagnosis Code 1 - 30 (DIAGNOSIS)	CHAR(36)	DX_CCEJ, J=1-30	DIAG_NOMEN	Apply format $nomen. to Diag_Nomen. Transpose records by Encounter_NK with formatted Diag_Type_Ref (format = $diag_type., code set = ‘17’) = 'Final' and contributor_system_dx = ’Inbound from CCE’. Sorted by PRIORITY_SEQ to determine order.
Contributor System (Diagnosis) (NOT RETAINED)	CHAR(16)	CONTRIBUTOR_SYSTEM_DX	CONTRIBUTOR_SYSTEM_REF	Apply format $contributor_system. (code set = 89)
Priority Sequence (Diagnosis)	N(8)	PRIORITY_SEQ	PRIORITY_SEQ	No transformation.

Field	Format	SAS Name	Source Element	Transformation
(NOT RETAINED)				
Clinical Diagnosis Priority (Diagnosis) (NOT RETAINED)	N(8)	CLINICAL_DIAG_PRIORITY	CLINICAL_DIAG_PRIORITY	No transformation.
PowerChart Diagnosis Code 1 - 50 (DIAGNOSIS)	CHAR(36)	DX_PCJ, J=1-50	DIAG_NOMEN	Apply format $nomen. to Diag_Nomen. Transpose records by Encounter_NK with formatted Diag_Type_Ref (format = $diag_type., code set = '17') = 'Discharge Diagnosis', 'Final' or 'Admitting Diagnosis' and contributor_system_dx = 'PowerChart'. Sorted by CLINICAl_DIAG_PRIORITY to determine order.
CERNER PROCEDURE (prdi_procedure)				
Anesthesia Minutes	N(8)	ANES_MINUTES	ANES_MINUTES	No transformation.
Priority Sequence	N(8)	PRIORITY_SEQ_PROC	PRIORITY_SEQ	No transformation.
Procedure Code (NOT RETAINED)	CHAR(7)	PROC	PROC_NOMEN	PROC = put(proc_nomen,$nomen)
Procedure File Primary Key	CHAR(100)	PROCEDURE_SK	PROCEDURE_SK	No transformation.
Units of Service (NOT RETAINED)	N(8)	PROCEDURE_DAY_CNT	PROCEDURE_DAY_CNT	No transformation.
MHS DMHRSi-HR				
Primary Care Manager (PCM) Location CD	CHAR(4)	PCMLOC	PROVMTFNM	No transformation.
Provider Assigned MEPRS Code For Additional Prov 1 (DMHRSi)	CHAR(4)	PROV_MEPRSD2	PROVMEPRNM	No transformation.
Provider Assigned MEPRS Code For Additional Prov 2 (DMHRSi)	CHAR(4)	PROV_MEPRSD3	PROVMEPRNM	No transformation.
Provider Assigned MEPRS Code For Additional Prov 3 (DMHRSi)	CHAR(4)	PROV_MEPRSD4	PROVMEPRNM	No transformation.
Provider Assigned MEPRS Code For Additional Prov 4 (DMHRSi)	CHAR(4)	PROV_MEPRSD5	PROVMEPRNM	No transformation.
Provider Assigned MEPRS Code For Additional Prov 5 (DMHRSi)	CHAR(4)	PROV_MEPRSD6	PROVMEPRNM	No transformation.
Provider Assigned MEPRS Code For Admitting Physician (DMHRSi)	CHAR(4)	PROV_MEPRSD_ADM	PROVMEPRNM	No transformation.
Provider Assigned MEPRS Code For Appointment Provider (DMHRSi)	CHAR(4)	PROV_MEPRSD1	PROVMEPRNM	No transformation.

Field	Format	SAS Name	Source Element	Transformation
Provider Assigned MEPRS Code For Attending Physician (DMHRSi)	CHAR(4)	PROV_MEPRSD_ATT	PROVMEPRNM	No transformation.
Provider Assigned MEPRS Code For Referring Physician (DMHRSi)	CHAR(4)	PROV_MEPRSD_REF	PROVMEPRNM	No transformation.
Provider Assigned MTF For Additional Provider 1 (DMHRSi)	CHAR(4)	PROV_MTFD2	PROVMTFNM	No transformation.
Provider Assigned MTF For Additional Provider 2 (DMHRSi)	CHAR(4)	PROV_MTFD3	PROVMTFNM	No transformation.
Provider Assigned MTF For Additional Provider 3 (DMHRSi)	CHAR(4)	PROV_MTFD4	PROVMTFNM	No transformation.
Provider Assigned MTF For Additional Provider 4 (DMHRSi)	CHAR(4)	PROV_MTFD5	PROVMTFNM	No transformation.
Provider Assigned MTF For Additional Provider 5 (DMHRSi)	CHAR(4)	PROV_MTFD6	PROVMTFNM	No transformation.
Provider Assigned MTF For Admitting Physician (DMHRSi)	CHAR(4)	PROV_MTFD_ADM	PROVMTFNM	No transformation.
Provider Assigned MTF For Appointment Provider (DMHRSi)	CHAR(4)	PROV_MTFD1	PROVMTFNM	No transformation.
Provider Assigned MTF For Attending Physician (DMHRSi)	CHAR(4)	PROV_MTFD_ATT	PROVMTFNM	No transformation.
Provider Assigned MTF For Referring Physician (DMHRSi)	CHAR(4)	PROV_MTFD_REF	PROVMTFNM	No transformation.
Provider Assigned Organization For Additional Provider 1 (DMHRSi)	CHAR(8)	PROV_ORGD2	PROVORGNM	No transformation.
Provider Assigned Organization For Additional Provider 2 (DMHRSi)	CHAR(8)	PROV_ORGD3	PROVORGNM	No transformation.
Provider Assigned Organization For Additional Provider 3 (DMHRSi)	CHAR(8)	PROV_ORGD4	PROVORGNM	No transformation.
Provider Assigned Organization For Additional Provider 4 (DMHRSi)	CHAR(8)	PROV_ORGD5	PROVORGNM	No transformation.
Provider Assigned Organization For Additional Provider 5 (DMHRSi)	CHAR(8)	PROV_ORGD6	PROVORGNM	No transformation.
Provider Assigned Organization For	CHAR(8)	PROV_ORGD_ADM	PROVORGNM	No transformation.

Field	Format	SAS Name	Source Element	Transformation
Admitting Physician (DMHRSi)				
Provider Assigned Organization For Appointment Provider (DMHRSi)	CHAR(8)	PROV_ORGD1	PROVORGNM	No transformation.
Provider Assigned Organization For Attending Physician (DMHRSi)	CHAR(8)	PROV_ORGD_ATT	PROVORGNM	No transformation.
Provider Assigned Organization For Referring Physician (DMHRSi)	CHAR(8)	PROV_ORGD_REF	PROVORGNM	No transformation.
Provider Assigned Service For Additional Provider 1 (DMHRSi)	CHAR(1)	PROV_SVC_ASSGD2	PROVSVCASSGNNM	No transformation.
Provider Assigned Service For Additional Provider 2 (DMHRSi)	CHAR(1)	PROV_SVC_ASSGD3	PROVSVCASSGNNM	No transformation.
Provider Assigned Service For Additional Provider 3 (DMHRSi)	CHAR(1)	PROV_SVC_ASSGD4	PROVSVCASSGNNM	No transformation.
Provider Assigned Service For Additional Provider 4 (DMHRSi)	CHAR(1)	PROV_SVC_ASSGD5	PROVSVCASSGNNM	No transformation.
Provider Assigned Service For Additional Provider 5 (DMHRSi)	CHAR(1)	PROV_SVC_ASSGD6	PROVSVCASSGNNM	No transformation.
Provider Assigned Service For Admitting Physician (DMHRSi)	CHAR(1)	PROV_SVC_ASSGD_ ADM	PROVSVCASSGNNM	No transformation.
Provider Assigned Service For Appointment Provider (DMHRSi)	CHAR(1)	PROV_SVC_ASSGD1	PROVSVCASSGNNM	No transformation.
Provider Assigned Service For Attending Physician (DMHRSi)	CHAR(1)	PROV_SVC_ASSGD_ ATT	PROVSVCASSGNNM	No transformation.
Provider Assigned Service For Referring Physician (DMHRSi)	CHAR(1)	PROV_SVC_ASSGD_ REF	PROVSVCASSGNNM	No transformation.
Provider Assigned Unit Identification Code (UIC) for Additional Provider 1 (DMHRSi)	CHAR(8)	PROV_UICD2	PROVUICNM	No transformation.
Provider Assigned Unit Identification Code (UIC) for Additional Provider 2 (DMHRSi)	CHAR(8)	PROV_UICD3	PROVUICNM	No transformation.
Provider Assigned Unit Identification Code (UIC) for Additional Provider 3 (DMHRSi)	CHAR(8)	PROV_UICD4	PROVUICNM	No transformation.
Provider Assigned Unit Identification Code (UIC) for Additional Provider 4 (DMHRSi)	CHAR(8)	PROV_UICD5	PROVUICNM	No transformation.

Field	Format	SAS Name	Source Element	Transformation
Provider Assigned Unit Identification Code (UIC) for Additional Provider 5 (DMHRSi)	CHAR(8)	PROV_UICD6	PROVUICNM	No transformation.
Provider Assigned Unit Identification Code (UIC) for Admitting Physician (DMHRSi)	CHAR(8)	PROV_UICD_ADM	PROVUICNM	No transformation.
Provider Assigned Unit Identification Code (UIC) for Appointment Provider (DMHRSi)	CHAR(8)	PROV_UICD1	PROVUICNM	No transformation.
Provider Assigned Unit Identification Code (UIC) for Attending Physician (DMHRSi)	CHAR(8)	PROV_UICD_ATT	PROVUICNM	No transformation.
Provider Assigned Unit Identification Code (UIC) for Referring Physician (DMHRSi)	CHAR(8)	PROV_UICD_REF	PROVUICNM	No transformation.
Provider Category For Additional Provider 1 (DMHRSi)	CHAR(22)	PROV_CATD2	PROVCATNM	No transformation.
Provider Category For Additional Provider 2 (DMHRSi)	CHAR(22)	PROV_CATD3	PROVCATNM	No transformation.
Provider Category For Additional Provider 3 (DMHRSi)	CHAR(22)	PROV_CATD4	PROVCATNM	No transformation.
Provider Category For Additional Provider 4 (DMHRSi)	CHAR(22)	PROV_CATD5	PROVCATNM	No transformation.
Provider Category For Additional Provider 5 (DMHRSi)	CHAR(22)	PROV_CATD6	PROVCATNM	No transformation.
Provider Category For Admitting Physician (DMHRSi)	CHAR(22)	PROV_CATD_ADM	PROVCATNM	No transformation.
Provider Category For Appointment Provider (DMHRSi)	CHAR(22)	PROV_CATD1	PROVCATNM	No transformation.
Provider Category For Attending Physician (DMHRSi)	CHAR(22)	PROV_CATD_ATT	PROVCATNM	No transformation.
Provider Category For Referring Physician (DMHRSi)	CHAR(22)	PROV_CATD_REF	PROVCATNM	No transformation.
Provider Service For Additional Provider 1 (DMHRSi)	CHAR(1)	PROV_SVCD2	PROVSVCNM	No transformation.
Provider Service For Additional Provider 2 (DMHRSi)	CHAR(1)	PROV_SVCD3	PROVSVCNM	No transformation.
Provider Service For Additional Provider 3 (DMHRSi)	CHAR(1)	PROV_SVCD4	PROVSVCNM	No transformation.

Field	Format	SAS Name	Source Element	Transformation
Provider Service For Additional Provider 4 (DMHRSi)	CHAR(1)	PROV_SVCD5	PROVSVCNM	No transformation.
Provider Service For Additional Provider 5 (DMHRSi)	CHAR(1)	PROV_SVCD6	PROVSVCNM	No transformation.
Provider Service For Admitting Physician (DMHRSi)	CHAR(1)	PROV_SVCD_ADM	PROVSVCNM	No transformation.
Provider Service For Appointment Provider (DMHRSi)	CHAR(1)	PROV_SVCD1	PROVSVCNM	No transformation.
Provider Service For Attending Physician (DMHRSi)	CHAR(1)	PROV_SVCD_ATT	PROVSVCNM	No transformation.
Provider Service For Referring Physician (DMHRSi)	CHAR(1)	PROV_SVCD_REF	PROVSVCNM	No transformation.
MHS MASTER PERSON INDEX (MPI)				
DEERS Patient Identifier (EDIPN)	CHAR(10)	EDIPN	EDIPN	No transformation.
DEERS Patient Social Security Number	CHAR(9)	PATSSN	SSN	No transformation.
Person Association Reason Code (PARC)	CHAR(2)	PARC	PARC	No transformation.
Sponsor Social Security Number	CHAR(9)	SPONSSN	SPSSN	No transformation.
MHS DEERS LONGITUDINAL (LVM)				
Alternate Care Value (ACV)	CHAR(1)	ACV	ACV	Blank for all records after Jan 1, 2018. If ACV is blank, then ACV = 'Z'
Alternate Care Value (ACV) Group	CHAR(2)	ACVGROUP	ACVGRP	Blank for all records after Jan 1, 2019. If ACVGROUP is blank, then ACVGROUP = 'O'
Assigned HCDP Code	CHAR(3)	HCDP_ASSGN	ASGHCDP	No transformation.
DEERS Beneficiary Category	CHAR(3)	BENCAT	BENCAT	If BENCAT is blank or 'Z' then BENCAT = 'UNK'
DEERS Common Beneficiary Category	CHAR(1)	COMBEN	CBEN	No transformation.
DEERS Eligibility Group	CHAR(1)	ELG_GRP	ELGGRP	Blank for all records before Jan 1, 2018. If ELG_GRP is blank, then ELG_GRP = 'Z'
DEERS Enrollment Group	CHAR(1)	ENR_GRP	ENRGRP	Blank for all records before Oct 1, 2017. If ENR_GRP is blank, then ENR_GRP = 'Z'
DEERS Marital Status	CHAR(1)	MARITAL	MS	No transformation.
DEERS Patient Date of Birth	CHAR(8)	PATDOB	DOB DOB_R	If blank, then use DOB_R. Format as yymmddn8.
DEERS Patient Gender	CHAR(1)	GENDER	GENDER	No transformation.
DEERS Patient Race Code	CHAR(1)	RACE	RACE	If RACE is blank, then RACE = 'Z'
DEERS Patient Zip Code	CHAR(5)	PATZIP	ZIP	No transformation.

Field	Format	SAS Name	Source Element	Transformation
DEERS Sponsor Service	CHAR(1)	SSVCLVM	SVC	No transformation.
DEERS Sponsor Service, Aggregated	CHAR(1)	SAGGLVM	AGGSVC	No transformation.
Enrollment MTF	CHAR(4)	ENR_MTF	ENR	If ENR_MTF is blank, then ENR_MTF = 'NONE'
Ethnic Background Code	CHAR(1)	ETHNIC	ETHNIC	If ETHNIC is blank, then ETHNIC = 'Z'
HCDP Code	CHAR(3)	HCDPLVM	HCDPLVM	No transformation.
Medicare Flag	CHAR(1)	MEDICARE_FLAG	MF	If MEDICARE_FLAG is blank, then MEDICARE_FLAG = 'N'
Patient Privilege Code	CHAR(1)	PRIVILEGE	PRIV	If PRIVILEGE is blank, then PRIVILEGE = '9'
Primary Care Manager (PCM) Provider ID	CHAR(18)	PCMIDLVM	PCM_ID	No transformation.
Primary Care Manager (PCM) Type	CHAR(1)	PCM_TYPE	PCMTYP	If PCM_TYPE is blank, then PCM_TYPE = 'Z'
TPR Eligibility Flag	CHAR(1)	TPRELIG	TPR	No transformation.
TRICARE Young Adult Flag	CHAR(1)	TYAFLAG	TYA	No transformation.
MHS CLIENT GROUPING SOFTWARE (CGS)				
MS-DRG Assigned MDC	CHAR(2)	MSMDC	Multiple	See Appendix B.
MS-DRG Code	CHAR(3)	MSDRG	Multiple	See Appendix B. Format as Z3.
MS-DRG Return Code	CHAR(2)	MSDRGRTC	Multiple	See Appendix B. Format as Z2.
MS-DRG Version	CHAR(4)	MSDRGVRSN	Multiple	See Appendix B.
Present on Admission (POA) Indicator 1-20	CHAR(1)	DX1POA-DX20POA		See Appendix B.NOTE: the POA variables are currently blank because they are not populated in the BDE.
MHS RWP WEIGHTS FORMAT FILE				
MS-DRG Baseline Relative Weighted Product	N(8)	MSDRGBASERWP	Multiple	MS-DRG baseline weight for the FY of record. See Appendix C for algorithm.
MS-DRG Transfer Status Flag	N(8)	MSDRGICAT	Multiple	Transfer indicator assigned in MS-DRG computation. See Appendix C for algorithm.
MS-DRG Outlier Relative Weighted Product	N(8)	MSDRGOUTRWP	Multiple	The increase or decrease in MS-DRG RWP resulting from long-stay or short-stay outliers or transfers. See Appendix C for algorithm.
MS-DRG Relative Weighted Product	N(8)	MSDRGRWP	Multiple	MSDRGBASERWP + MSDRGOUTRWP See Appendix C for algorithm. For inferred records, merge to reference file by MTF and MEPR3 and assign value (no multiplication).
MHS MEDICAL / SURGICAL REFERENCE FILE				

Field	Format	SAS Name	Source Element	Transformation
MS-DRG Medical/Surgical Indicator	CHAR(1)	MSDRGSURG	MSDRGSURG	If MSDRGSURG=1, then MSDRGSURG='S' Else MSDRGSURG='M'
MHS DMIS ID INDEX FILE				
Enrollment MTF Branch of Service	CHAR(1)	ENR_SVC	UBU_SVC	No transformation.
Enrollment MTF T17 Region	CHAR(2)	ENR_T17_REG	T17_REG	No transformation.
Enrollment MTF T3 Region	CHAR(2)	ENR_T3_REG	T3_REG	No transformation.
Enrollment Parent MTF	CHAR(4)	ENR_PARENT	UBU_PAR	No transformation.
PPS Enrollment Site Parent	CHAR(4)	ENR_PPS_PARENT	ENR_MTF	No transformation.
PPS Treatment MTF Parent	CHAR(4)	PPS_PARENT	MTF	No transformation.
Treatment MTF Branch of Service	CHAR(1)	MTF_SVC	UBU_SVC	No transformation.
Treatment MTF Command	CHAR(8)	MTF_CMND	MAJCMND	No transformation.
Treatment MTF MSMA	CHAR(3)	MTF_MSMA	MSM_ID	No transformation.
Treatment MTF T17 Region	CHAR(2)	MTF_T17_REG	T17_REG	No transformation.
Treatment MTF T3 Region	CHAR(2)	MTF_T3_REG	T3_REG	No transformation.
Treatment Parent MTF (DMISID Index File) (NOT RETAINED)	CHAR(4)	MTF_PARENT_D	UBU_PAR	No transformation.
MHS OMNI-CAD				
Beneficiary Catchment Area	CHAR(4)	CATCH	WORLD	No transformation.
Beneficiary MTF Service Area	CHAR(4)	MTFSVCAREA	BPA	No transformation.
Beneficiary PRISM Area	CHAR(4)	PRISM	PRISM	No transformation.
Beneficiary T17 Region	CHAR(2)	BEN_T17_REG	T17REG	No transformation.
Beneficiary T3 Region	CHAR(2)	BEN_T3_REG	REGION	No transformation.
DERIVED				
Admission Date & Time	N(8)	ADM_DT_TM	ADMIT_DT_TM INPATIENT_ADMIT_DT_TM PRE_ADMIT_DT_TM	Run macro %local_offset using DT_TM & TM_ZN variables for each date to get _LOCAL variables. Apply format e8601dt. Use INPATIENT_ADMIT_DT_TM unless blank. If blank, use ADMIT_DT_TM. If both are blank, use PRE_ADMIT_DT_TM
Admission Inferred Flag (MHS GENESIS)	N(8)	ADM_INFR_FLAG	DISCHARGE_DT_TM DX_CCE1	1, if DISCHARGE_DT_TM or DX_CCE1 is missing. 0, otherwise.
Admission Location Specialty Type	CHAR(10)	ADMISSION_LOC_COMPSPEC	ADMISSION_LOC_COMPOSITE	Blank, if ADMISSION_LOC_COMPOSITE = 0 Else, scan(ADMISSION_LOC_COMPOSITE,2,'-')

Field	Format	SAS Name	Source Element	Transformation
Admission Location Type of Care	CHAR(10)	ADMISSION_LOC_COMPCARE	ADMISSION_LOC_COMPOSITE	Blank, if ADMISSION_LOC_COMPOSITE=0 Else, scan(ADMISSION_LOC_COMPOSITE,3,'-')
Admission Source, Legacy	CHAR(43)	ADMIT_SOURCE_LEGACY	ADMIT_SOURCE ADMIT_TYPE COMBEN	0, if ADMIT_TYPE = Emergency & ADMIT_SOURCE in (Clinic Referral, Patient/Self, Not Available) L, else if ADMIT_SOURCE = Born Inside the Hospital S, else if ADMIT_SOURCE = Transfer from Ambulatory Surgery Center 1, else if ADMIT_SOURCE in (Clinic Referral, Court/Law Enforcement, Non-Health Care Facility Source of Origin, Patient/Self, Born Outside This Hospital, Not Available) 4, else if ADMIT_SOURCE is not blank & COMBEN = 4 5, else if ADMIT_SOURCE is not blank Blank, otherwise.
Age Group	CHAR(1)	AGEGRP	PATAGE	%age_group_dmis(patage,AGEGRP)
Age Group Common	CHAR(1)	EXPAGE	PATAGE	Derived from PATAGE: A = 0-4 B = 5-14 C = 15-17 D = 18-24 E = 25-34 F = 35-44 G = 45-64 H = 65-69 I = 70-74 J = 75-79 K = 80-84 L = 85+ X = All others
AHRQ Indicator: Adult Short Term Diabetes Complications	Char(1)	ASTDIAB	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1	If MSMDC = 14 then astdiab = 0. If PATAGE <18 then astdiab = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then astdiab = 0. If primary diagnosis is not in the format $ACDIASD then astdiab = 0. Else if primary diagnosis is in $ACDIASD then astdiab = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Perforated Appendix	Char(1)	APAPPD	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1 – DX20	If MSMDC = 14 then apappd = 0. If PATAGE < 18 then apappd = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then apappd = 0. If any diagnosis is in format $ACSAP2D and not in format $ACSAPPD then apappd = 2. Else if any diagnosis is in $ACSAPPD then apappd = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Diabetes Long Term Complications	Char(1)	ALTDIAB	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1	If MSMDC = 14 then altdiab = 0. If PATAGE < 18 then altdiab = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then altdiab = 0. If primary diagnosis is not in format $ACDIALD then altdiab = 0. Else if primary diagnosis is in format $ACDIALD then altdiab = 1.

Field	Format	SAS Name	Source Element	Transformation
				Not populated for E Admissions.
AHRQ Indicator: Adult Chronic Obstructive Pulmonary Disorder	Char(1)	ACOPD	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1 – DX20	If MSMDC = 14 then acopd = 0. If PATAGE < 40 then acopd = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then acopd = 0. If primary diagnosis is in format $ACCOPDD or primary diagnosis is in format $ACCPD2D and any secondary diagnoses are in format $ACCOPDD then acopd = 1. Else acopd = 0. Not populated for E Admissions.
AHRQ Indicator: Adult Hypertension Admission	Char(1)	AHYPTN	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If MSMDC = 14 then ahyptn = 0. If PATAGE < 18 then ahyptn = 0. If any procedure code is in format $ACSCARP then ahyptn = 0. If any procedure code is in format $ACSHYPP then ahyptn = 0. If any diagnosis is in format $ACSHY2D then ahyptn = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then ahyptn = 0. If primary diagnosis is not in format $ACSHYPD then ahyptn = 0. Else if primary diagnosis is in format $ACSHYPD then ahyptn = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Congestive Heart Failure Admission	Char(1)	ACHF	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If MSMDC = 14 then achf = 0. If PATAGE < 18 then achf = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then achf = 0. If any procedure code is in format $ACSCARP then achf = 0. If primary diagnosis is not in format $ACSCH2D then achf = 0. Else if primary diagnosis is in format $ACSCHFD then achf = 1. Not populated for E Admissions.
AHRQ Indicator: Low Birth Weight	Char(1)	ALBW	PATAGE (in days) PATAGE (in years) ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	See Appendix G. Not populated for E Admissions.
AHRQ Indicator: Adult Dehydration	Char(1)	ADHYD	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1 – DX20	If MSMDC = 14 then adhyd = 0. If PATAGE < 18 then adhyd = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then adhyd = 0. If any diagnosis is in format $CRENLFD then adhyd = 0. If primary diagnosis is not in format $ACSDEHD then adhyd = 0. Else If primary diagnosis is in format $ACSDEHD or primary diagnosis is in $HYPERID, $ACPGASD, or $PHYSIDB and any secondary diagnosis is in $ACSDEHD then adhyd = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Bacterial Pneumonia	Char(1)	ABACPN	MSMDC PATAGE	If MSMDC = 14 then abacpn = 0. If PATAGE < 18 then abacpn = 0.

Field	Format	SAS Name	Source Element	Transformation
			ADMIT_SOURCE_LEGACY DX1 – DX20	If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then abacpn = 0. If any diagnosis is in format $ACSBA2D then abacpn = 0. If any diagnosis is in format $IMMUNID then abacpn = 0. If any procedure is in format $IMMUNIP then abacpn = 0. If primary diagnosis is not in format $ACSBACD then abacpn = 0. Else if primary diagnosis is in $ACSBACD then abacpn = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Urinary Tract Infection	Char(1)	AUTI	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1 – DX20 PROC_1-PROC_20	If MSMDC = 14 then auti = 0. If PATAGE < 18 then auti = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then auti = 0. If any diagnosis is in format $IMMUNID then auti = 0. If any diagnosis is in format $KIDNEY then auti = 0. If any procedure is in format $IMMUNIP then auti = 0. Else if primary diagnosis is in format $ACSUTID then auti = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Angina without Procedure	Char(1)	AAWP	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1	If MSMDC = 14 then aawp = 0. If PATAGE < 18 then aawp = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then aawp = 0. If any procedure code is in format $ACSCARP then aawp = 0. If primary diagnosis is not in format $ACSANGD then aawp = 0. Else if primary diagnosis is in format $ACSANGD then aawp = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Uncontrolled Diabetes	Char(1)	AUNCDIAB	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1	If MSMDC = 14 then auncdiab = 0. If PATAGE < 18 then auncdiab = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then auncdiab = 0. If Primary diagnosis is not in format $ACDIAUD then auncdiab = 0. Else if primary diagnosis is in format $ACDIAUD then auncdiab = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Asthma	Char(1)	AASTH	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1 – DX20	If MSMDC = 14 then aasth = 0. If PATAGE < 18 then aasth = 0. If any diagnosis is in format $RESPAN then aasth = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then aasth = 0. If primary diagnosis is not in format $ACSASTD then aasth = 0. Else if primary diagnosis is in format $ACSASTD then aasth = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Lower-extremity	Char(1)	AAMPDIADX	MSMDC PATAGE ADMIT_SOURCE_LEGACY	If MSMDC = 14 then aampdiab = 0. If PATAGE < 18 then aampdiab = 0.

Field	Format	SAS Name	Source Element	Transformation
Amputation among patients with Diabetes			PROC_1-PROC_20 DX1 – DX20	If any diagnosis codes are in format $ACLEA2D then aampdiab = 0. If any procedure is in format $TOEAMIP then aampdiab = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then aampdiab = 0. Else if any diagnosis codes are in format $ACSLEAD and any procedure codes are format $ACSLEAP then aampdiab = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Overall Composite	Char(1)	AOVALL	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If astdiab = 1 or altdiab = 1 or acopd = 1 or ahyptn = 1 or achf = 1 or adhyd = 1 or abacpn = 1 or auti = 1 or aawp = 1 or auncdiab = 1 or aasth = 1 or aampdiab = 1 then aovall = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Acute Composite	Char(1)	AACUTE	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If adhyd = 1 or abacpn = 1 or auti = 1 then aacute = 1. Not populated for E Admissions.
AHRQ Indicator: Adult Chronic Composite	Char(1)	ACHRON	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If astdiab = 1 or altdiab = 1 or acopd = 1 or ahyptn = 1 or achf = 1 or aawp = 1 or auncdiab = 1 or aasth = 1 or aampdiab = 1 then achron = 1. Not populated for E Admissions.
AHRQ Indicator: Pediatric Asthma Admission	Char(1)	PASTH	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If MSMDC = 14 then pasth = 0. If PATAGE < 2 then pasth = 0. If PATAGE > 17 then pasth = 0. If any diagnosis is in format $RESPAN then pasth = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then pasth = 0. If primary diagnosis is not in format $ACSASTD then pasth = 0. Else if primary diagnosis is in format $ACSASTD then pasth = 1. Not populated for E Admissions.
AHRQ Indicator: Pediatric Short term Diabetes	Char(1)	PSTDIAB	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1	If MSMDC = 14 then pstdiab = 0. If PATAGE < 6 then pstdiab = 0. If PATAGE > 17 then pstdiab = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then pstdiab = 0. If primary diagnosis is not in $ACDIASD then pstdiab = 0. Else if primary diagnosis is in $ACDIASD then pstdiab = 1. Not populated for E Admissions.
AHRQ Indicator: Pediatric Gastroenteritis	Char(1)	PGASTRO	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1 – DX20	MSMDC = 14 then pgastro = 0. If PATAGE < 3 months pgastro = 0. If PATAGE > 17 then pgastro = 0. If any diagnosis codes are in format $ACGDISD then pgastro = 0. If any diagnosis code is in format $ACBACGD then pgastro = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then pgastro = 0. If primary diagnosis is in format $ACPGASD then pgastro = 1.

Field	Format	SAS Name	Source Element	Transformation
				If primary diagnosis is in $ACSDEHD and any secondary diagnosis is in $ACPGASD then pgastro =1. Else pgastro = 0. Not populated for E Admissions.
AHRQ Indicator: Pediatric Perforated Appendix	Char(1)	PAPPD	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1 – DX20	If MSMDC = 14 then pappd = 0. If PATAGE < 1 then pappd = 0. If PATAGE > 17 then pappd = 0. If ADMIT_SOURCE_LEGACY=4,5,6,7,8 then pappd = 0. If any diagnosis is in format $ACSAP2D and no diagnosis is in $ACSPPD then pappd = 2. If any diagnosis is in $ACSAPPD then pappd = 1. Else pappd = 0. Not populated for E Admissions.
AHRQ Indicator: Pediatric Urinary Tract Infection	Char(1)	PUTI	MSMDC PATAGE ADMIT_SOURCE_LEGACY DX1 – DX20 PROC_1-PROC_20	If MSMDC = 14 then puti = 0. If PATAGE < 3 months then puti = 0. If PATAGE > 17 then puti = 0. If any diagnosis is in format $IMMUNHD then puti = 0. If any diagnosis is in format $KIDNEY then puti = 0. If any diagnosis is in format $IMMUITD then puti = 0. If any diagnosis is in format $HEPFA2D then puti = 0. If any diagnosis is in format $HEPFA3D then puti = 0. If any procedure is in format $TRANSPP then puti = 0. Else if primary diagnosis is in format $ACSUTID then puti = 1. Not populated for E Admissions.
AHRQ Indicator: Pediatric Overall Composite	Char(1)	POVALL	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If pasth=1 or pstdiab=1 or pgastro=1 or puti=1 then povall=1. Else povall=0. Not populated for E Admissions.
AHRQ Indicator: Pediatric Chronic Composite	Char(1)	PCHRON	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	IF pasth=1 or pstdiab=1 then pchron=1. Else pchron=0. Not populated for E Admissions.
AHRQ Indicator: Pediatric Acute Composite	Char(1)	PACUTE	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If pgastro=1 or puti=1 then pacute=1. Else pacute=0. Not populated for E Admissions.
AHRQ Indicator: Combined Overall Adult and Pediatric Composite	Char(1)	PADCDOVL	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If aovall=1 or povall=1 then padcdovl=1. Else padcdovl=0. Not populated for E Admissions.
AHRQ Indicator: Combined Chronic Adult and Pediatric Composite	Char(1)	PADCDCHN	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If achron=1 or pchron=1 then padcdchn=1. Else padcdchn=0. Not populated for E Admissions.

Field	Format	SAS Name	Source Element	Transformation
AHRQ Indicator: Combined Acute Adult and Pediatric Composite	Char(1)	PADCDACT	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If aacute=1 or pacute=1 then padcdact=1. Else padcdact=0. Not populated for E Admissions.
AHRQ Indicator: AHRQ Prevention Indicator Flag	Char(1)	AHRQPVADM	MSMDC PATAGE ADMIT_SOURCE_LEGACY PROC_1-PROC_20 DX1 – DX20	If astdiab = 1 then ahrqpvadm = A. If apappd = 1 then ahrqpvadm = B. If altdiab = 1 then ahrqpvadm = C. If acopd = 1 then ahrqpvadm = D. If ahyptn = 1 then ahrqpvadm = E. If achf = 1 then ahrqpvadm = F. If albw = 1 then ahrqpvadm = G. If adhyd = 1 then ahrqpvadm = H. If abacpn = 1 then ahrqpvadm = I. If auti = 1 then ahrqpvadm = J. If aawp = 1 then ahrqpvadm = K. If auncdiab = 1 then ahrqpvadm = L. If aasth = 1 then ahrqpvadm = M. If aampdiab = 1 then ahrqpvadm = N. If pasth = 1 then ahrqpvadm = P. If pstdiab = 1 then ahrqpvadm = Q. If pgastro = 1 then ahrqpvadm = R. If pappd = 1 then ahrqpvadm = S. If puti = 1 then ahrqpvadm = T. Else ahrqpvadm = O. Not populated for E Admissions.
Appointment Prefix (Source System Flag)	CHAR(1)	APPTPFIX		Set to M for all GENESIS records, blank otherwise.
Bed Days from Segments (Enc History)	N(8)	BED_DAYS	DAYS_1 – DAYS_30	Sum of DAYS_1 to DAYS_30.
Bed Days from Encounter (Enncounter)	N(8)	MTF_DAYS	ADM_DT_TM DISCHARGE_DT_TM	Datepart(DISCHARGE_DT_TM) – datepart(ADM_DT_TM)
Beneficiary First Name	CHAR(20)	FIRSTNAME	MPI Merge FIRSTNAME PERSON File FIRST_NAME	Use First Name from MPI Merge, if available. Otherwise, use First Name from Cerner data in the PERSON file.
Beneficiary Last Name	CHAR(26)	LASTNAME	MPI Merge LASTNAME PERSON File LAST_NAME	Use Last Name from MPI Merge, if available. Otherwise, use Last Name from Cerner data in the PERSON file.
Beneficiary Patient Name	CHAR(74)	PATNAME	LASTNAME FIRSTNAME	Concatenate(LASTNAME,',', FIRSTNAME)
Birthweight	N(8)	BIRTH_WEIGHT	Multiple	See Appendix F.
Birthweight Source	N(8)	BW_SOURCE	Multiple	See Appendix F.
Calculated Bed Days from Encounter (Enncounter)	N(8)	CALCDAYS	MTF_DAYS	MTF_DAYS, if MTF_DAYS > 0 1, if MTF_DAYS = 0
Calendar Month, Admission	CHAR(2)	CMADM	ADM_DT_TM	put(month(datepart(adm_dt_tm)),4.)
Calendar Year, Admission	CHAR(4)	CYADM	ADM_DT_TM	put(year(datepart(adm_dt_tm)),z2.)
Calendar Month, Discharge	CHAR(2)	CMDISP	DISCHARGE_DT_TM	put(month(datepart(discharge_dt_tm)),4.)

Field	Format	SAS Name	Source Element	Transformation
Calendar Year, Discharge	CHAR(4)	CYDISP	DISCHARGE_DT_TM	put(year(datepart(discharge_dt_tm)),z2.)
CCE Completed Date & Time	N(8)	CCE_COMPLETED_DT_TM	CODING_COMPLETED_DT_TM CODING_COMPLETED_TM_ZN	Run macro %local_offset using DT_TM & TM_ZN variables for each date to get _LOCAL variables. Rename to final. Apply format e8601dt.
Departure Date & Time	N(8)	DEPART_DT_TM	DEPART_DT_TM DEPART_TM_ZN	Run macro %local_offset using DT_TM & TM_ZN variables for each date to get _LOCAL variables. Rename to final. Apply format e8601dt.
Diagnosis Code 1 - 20	CHAR(36)	DX*J*	DX_CCE*J*, *J*=1 to 20 DX_PC*J*, *J*=1 to 20	DX*J* = DX_CCE*J*, if DX_CCE*J* is not blank. Else DX*J* = DX_PC*J* if DX_PC*J* is not blank. Otherwise, DX*J* is blank. Keep the 1st 20 diagnoses.
Diagnosis Group	CHAR(2)	DXGRP	DX1	Use first three characters of Diagnosis 1 (DX1). See Table D2 for derivation rules.
Discharge Date & Time	N(8)	DISCHARGE_DT_TM	DISCHARGE _DT_TM DISCHARGE _TM_ZN	Run macro %local_offset using DT_TM & TM_ZN variables for each date to get _LOCAL variables. Rename to final. Apply format e8601dt.
Discharge Date	CHAR(8)	DISPDATE	DISCHARGE _DT_TM	put(datepart(DISCHARGE_DT_TM),yymmddn8.) Apply format yymmddn8.
Disposition Code, Legacy	CHAR(1)	DISPCODE_LEGACY	DISCHARGE_DISPOSITION_REF	01, If dispcode in ('Home' 'Advice Assessment' 'Released Without Limitations' 'Returned to Duty' 'Sick at Home/Quarters' 'Results Received' 'Released With Work Duty Limitations') 02, if dispcode in ('Discharged to Civilian Facility' 'Transfer to Another Hospital' 'Discharged to Joint MTF' 'Transfer to Army MTF' 'Transfer to Short Term Facility' 'Transfer to Navy MTF') 03, if dispcode in ('Transfer to SNF') 07, if dispcode in ('Against Medical Advise' 'Elopement' 'Left Without Being Seen') 20, if dispcode in ('Patient has expired') 30, if dispcode in ('Admitted as inpatient' 'Continued Stay' 'Transfer to Another Clinical Service') 43, if dispcode in ('Discharged to Other Federal Facility') 62, if dispcode in ('Discharged to ICF') 70, if dispcode in ('Transfer to Other') 72, if dispcode in ('Place in Observation' 'Referred to ER') Blank, otherwise
During Stay Flag	N(8)	DURING_STAY_FLAG	ADM_DT_TM DISCHARGE_DT_TM TRANSACT_DT_TM	See Appendix E.
Facility Admit Date & Time	N(8)	ADMIT_DT_TM	ADMIT_DT_TM ADMIT_TM_ZN	Run macro %local_offset using DT_TM & TM_ZN variables for each date to get _LOCAL variables. Rename to final. Apply format e8601dt.
Facility Arrival Date & Time	N(8)	ARRIVE_DT_TM	ARRIVE_DT_TM ARRIVE_TM_ZN	Run macro %local_offset using DT_TM & TM_ZN variables for each date to get _LOCAL variables. Rename to final. Apply format e8601dt.

Field	Format	SAS Name	Source Element	Transformation
Fiscal Month, Admission	CHAR(2)	FMADM	CMADM	If CMADM in ('10' '11 '12') then FMADM = put(CMADM - 9,z2.). Else FM = put(CMADM + 3, z2.).
Fiscal Year, Admission	CHAR(4)	FYADM	CMADM CYADM	If CMADM in ('10' '11 '12') then FYADM = put(CYADM+1,4.). Else FYADM = CYADM.
Fiscal Month, Discharge	CHAR(2)	FMDISP	CMDISP	If CMDISP in ('10' '11 '12') then FMDISP = put(CMDISP - 9,z2.). Else FM = put(CMDISP + 3, z2.).
Fiscal Year, Discharge	CHAR(4)	FYDISP	CMDISP CYDISP	If CMDISP in ('10' '11 '12') then FYDISP = put(CYDISP+1,4.). Else FYDISP = CYDISP.
Follow Up Flag	N(8)	FOLLOW_UP_FLAG	DISCHARGE_DT_TM TRANSACT_DT_TM	See Appendix E.
GENESIS Flag	N(8)	GENESIS_FLAG		Set to 1 if record is from MHS GENESIS, Else set to 0.
ICU Days (Enc History)	N(8)	ICU_DAYS	DAYS_1 – DAYS_30 NURSELOC_1- NURSELOC_30	See Appendix E.
Inpatient Admit Date & Time (Procedures)	N(8)	INPATIENT_ADMIT_DT_TM	INPATIENT_ADMIT_DT_TM INPATIENT_ADMIT_TM_ZN	Run macro %local_offset using DT_TM & TM_ZN variables for each date to get _LOCAL variables. Rename to final. Apply format e8601dt.
Legacy Discharge Disposition Code	CHAR(2)	DISPCODE_LEGACY	DISCHARGE_DISPOSITION_REF	All values blank
Medical Record Number (Patient)	CHAR(40)	MRN	FORMATTED_MRN (FROM ENCOUNTER) MRNPERSON (FROM PERSON)	Use value from Encounter File unless blank. If blank use the value from the variable MRN in the Person file.
Observation Flag	CHAR(1)	OBS_FLAG	DX1 MTF_DAYS	Y, when DX1='Z049' and MTF_DAYS<2 N, otherwise
Patient Age	N(8)	PATAGE	PATDOB DOB_R ADM_DT_TM	If PATDOB is not missing then: %age(patdob, datepart(adm_dt_tm),years=PATAGE) Else if DOB_R is not missing then: %age(datepart(dob_r),datepart(adm_dt_tm),years=PATAGE)
Patient Gender	CHAR(1)	PATSEX	GENDER GENDER_R	PATSEX = GENDER. If PATSEX = ' ' and GENDER_R is 'F' or 'M' then PATSEX = GENDER_R.
Patient OHI Flag	CHAR(1)	OHI	FINANCIAL_CLASS_REF	Apply format $financial_class. (code set = 354) OHI = 1 when financial_class is one of the following: Preferred Provider Organization (PPO) Blue Cross/Blue Shield Commercial/OHI HMO OHI = 0 otherwise. This includes the following values of financial_class: Tricare Medicare Other Federal Program Self Pay
PPS Earnings Factor	N(5,3)	PPS_EARNINGS_FACTOR		Assign value as 1.000
PPS Rate Basis	CHAR(1)	PPS_RATE_BASIS	MSMDC	D, If MSMDC = 19 or 20 R, otherwise

Field	Format	SAS Name	Source Element	Transformation
Preadmit Visit Flag	N(8)	PREADMIT_FLAG	ADM_DT_TM TRANSACT_DT_TM ENCOUNTER_TYPE	See Appendix E.
Pre Admit Date & Time (Procedures)	N(8)	PRE_ADMIT_DT_TM	PRE_ADMIT_DT_TM PRE_ADMIT_TM_ZN	Run macro %local_offset using DT_TM & TM_ZN variables for each date to get _LOCAL variables. Rename to final. Apply format e8601dt.
Primary HIPAA Taxonomy For Primary Provider	CHAR(10)	PROV_HIPAA_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_HIPAA_ADM PROV_HIPAA_ATT PROV_HIPAA1 PROV_HIPAA2	If PROVID_ATT is not blank, then use PROV_HIPAA_ATT Else if PROVID1 is not blank, then use PROV_HIPAA1 Else if PROVID_ADM is not blank, then use PROV_HIPAA_ADM Else if PROVID2 is not blank, then use PROV_HIPAA2. Else blank.
Primary Provider Source	CHAR(4)	PROV_PRIM_SOURCE	PROVID_ADM PROVID_ATT PROVID1 PROVID2	If PROVID_ATT is not blank, then use 'ATT' Else if PROVID1 is not blank, then use 'APPT' Else if PROVID_ADM is not blank, then use 'ADM' Else if PROVID2 is not blank, then use 'ADD1' Else blank.
Procedure Code	CHAR(7)	PROC_1-PROC_20	PROC ENCOUNTER_NK	Limit PROC to length 7 and transpose based on ENCOUNTER_NK. Keep the 1st 20.
Procedure Code Date & Time (Procedures)	N(8)	PROC_DT_TM	PROC_DT_TM PROC_TM_ZN	Run macro %local_offset using DT_TM & TM_ZN variables for each date to get _LOCAL variables. Rename to final. Apply format e8601dt.
Procedure Code Units of Service	N(8)	PROCQTY_1-PROCQTY_20	PROCEDURE_DAY_CNT ENCOUNTER_NK	If PROCEDURE_DAY_CNT=0 then change to =1. Transpose the modified PROCEDURE_DAY_CNT by ENCOUNTER_NK. Keep the 1st 20 codes.
Procedure Code Units of Service (RAW)	N(8)	PROCQTY_RAW_1-PROCQTY_RAW_20	PROCEDURE_DAY_CNT ENCOUNTER_NK	Transpose PROCEDURE_DAY_CNT by ENCOUNTER_NK. Keep the 1st 20 codes.
Product Line	CHAR(2)	PRODLINE	ADM_INFR_FLAG MSMDC MSDRGSURG	When ADM_INFR_FLAG = 0 then, OB, when MSMDC = 14 or 15 MH, when MSMDC = 19 or 20 M, when MSDRGSURG = M S, when MSDRGSURG = S
Provider Assigned Service For Primary Provider (DMHRSi)	CHAR(1)	PROV_SVC_ASSGD_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_SVC_ASSGD_ADM PROV_SVC_ASSGD_ATT PROV_SVC_ASSGD1 PROV_SVC_ASSGD2	If PROVID_ATT is not blank, then use PROV_SVC_ASSGD_ATT Else if PROVID1 is not blank, then use PROV_SVC_ASSGD1 Else if PROVID_ADM is not blank, then use PROV_SVC_ASSGD_ADM Else if PROVID2 is not blank, then use PROV_SVC_ASSGD2. Else blank.
Provider Assigned MEPRS For Primary Provider (DMHRSi)	CHAR(4)	PROV_MEPRSD_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_MEPRSD_ADM PROV_MEPRSD_ATT PROV_MEPRSD1 PROV_MEPRSD2	If PROVID_ATT is not blank, then use PROV_MEPRSD_ATT Else if PROVID1 is not blank, then use PROV_MEPRSD1 Else if PROVID_ADM is not blank, then use PROV_MEPRSD_ADM Else if PROVID2 is not blank, then use PROV_MEPRSD2. Else blank.

Field	Format	SAS Name	Source Element	Transformation
Provider Assigned MTF For Primary Provider (DMHRSi)	CHAR(4)	PROV_MTFD_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_MTFD_ADM PROV_MTFD_ATT PROV_MTFD1 PROV_MTFD2	If PROVID_ATT is not blank, then use PROV_MTFD_ATT Else if PROVID1 is not blank, then use PROV_MTFD1 Else if PROVID_ADM is not blank, then use PROV_MTFD_ADM Else if PROVID2 is not blank, then use PROV_MTFD2. Else blank.
Provider Assigned Organization For Primary Provider (DMHRSi)	CHAR(8)	PROV_ORGD_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_ORGD_ADM PROV_ORGD_ATT PROV_ORGD1 PROV_ORGD2	If PROVID_ATT is not blank, then use PROV_ORGD_ATT Else if PROVID1 is not blank, then use PROV_ORGD1 Else if PROVID_ADM is not blank, then use PROV_ORGD_ADM Else if PROVID2 is not blank, then use PROV_ORGD2. Else blank.
Provider Assigned UIC For Primary Provider (DMHRSi)	CHAR(8)	PROV_UICD_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_UICD_ADM PROV_UICD_ATT PROV_UICD1 PROV_UICD2	If PROVID_ATT is not blank, then use PROV_UICD_ATT Else if PROVID1 is not blank, then use PROV_UICD1 Else if PROVID_ADM is not blank, then use PROV_UICD_ADM Else if PROVID2 is not blank, then use PROV_UICD2. Else blank.
Provider Category For Primary Provider (DMHRSi)	CHAR(22)	PROV_CATD_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_CATD_ADM PROV_CATD_ATT PROV_CATD1 PROV_CATD2	If PROVID_ATT is not blank, then use PROV_CATD_ATT Else if PROVID1 is not blank, then use PROV_CATD1 Else if PROVID_ADM is not blank, then use PROV_CATD_ADM Else if PROVID2 is not blank, then use PROV_CATD2. Else blank.
Provider EDIPN For Primary Provider	CHAR(10)	PROV_EDIPN_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_EDIPN_ADM PROV_EDIPN_ATT PROV_EDIPN1 PROV_EDIPN2	If PROVID_ATT is not blank, then use PROV_EDIPN_ATT Else if PROVID1 is not blank, then use PROV_EDIPN1 Else if PROVID_ADM is not blank, then use PROV_EDIPN_ADM Else if PROVID2 is not blank, then use PROV_EDIPN2. Else blank.
Provider ID For Primary Provider	CHAR(100)	PROVID_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2	If PROVID_ATT is not blank, then use PROVID_ATT Else if PROVID1 is not blank, then use PROVID1 Else if PROVID_ADM is not blank, then use PROVID_ADM Else if PROVID2 is not blank, then use PROVID2. Else blank.
Provider NPI For Primary Provider	CHAR(10)	PROV_NPI_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_NPI_ADM	If PROVID_ATT is not blank, then use PROV_NPI_ATT Else if PROVID1 is not blank, then use PROV_NPI1

Field	Format	SAS Name	Source Element	Transformation
			PROV_NPI_ATT PROV_NPI1 PROV_NPI2	Else if PROVID_ADM is not blank, then use PROV_NPI_ADM Else if PROVID2 is not blank, then use PROV_NPI2. Else blank.
Provider NPI Type For Additional Provider 1	CHAR(1)	PROV_NPITYPE2		If PROVNPI2 is not blank then = 1 (Individual Provider)
Provider NPI Type For Additional Provider 2	CHAR(1)	PROV_NPITYPE3		If PROVNPI3 is not blank then = 1 (Individual Provider)
Provider NPI Type For Additional Provider 3	CHAR(1)	PROV_NPITYPE4		If PROVNPI4 is not blank then = 1 (Individual Provider)
Provider NPI Type For Additional Provider 4	CHAR(1)	PROV_NPITYPE5		If PROVNPI5 is not blank then = 1 (Individual Provider)
Provider NPI Type For Additional Provider 5	CHAR(1)	PROV_NPITYPE6		If PROVNPI6 is not blank then = 1 (Individual Provider)
Provider NPI Type For Admitting Physician	CHAR(1)	PROV_NPITYPE_ADM		If PROVNPI_ADM is not blank then = 1 (Individual Provider)
Provider NPI Type For Appointment Provider	CHAR(1)	PROV_NPITYPE1		If PROVNPI_ATT is not blank then = 1 (Individual Provider)
Provider NPI Type For Attending Physician	CHAR(1)	PROV_NPITYPE_ATT		If PROVNPI_ATT is not blank then = 1 (Individual Provider)
Provider NPI Type For Primary Provider	CHAR(1)	PROV_NPITYPE_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_NPITYPE_ADM PROV_NPITYPE_ATT PROV_NPITYPE1 PROV_NPITYPE2	If PROVID_ATT is not blank, then use PROV_NPITYPE_ATT Else if PROVID1 is not blank, then use PROV_NPITYPE1 Else if PROVID_ADM is not blank, then use PROV_NPITYPE_ADM Else if PROVID2 is not blank, then use PROV_NPITYPE2. Else blank.
Provider NPI Type For Referring Physician	CHAR(1)	PROV_NPITYPE_REF		If PROVNPI_REF is not blank then = 1 (Individual Provider)
Provider Role For Primary Provider	CHAR(40)	PROV_ROLE_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_ROLE_ADM PROV_ROLE_ATT PROV_ROLE1 PROV_ROLE2	If PROVID_ATT is not blank, then use PROV_ROLE_ATT Else if PROVID1 is not blank, then use PROV_ROLE1 Else if PROVID_ADM is not blank, then use PROV_ROLE_ADM Else if PROVID2 is not blank, then use PROV_ROLE2. Else blank.
Provider Service For Primary Provider (DMHRSi)	CHAR(1)	PROV_SVCD_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 PROV_SVCD_ADM PROV_SVCD_ATT PROV_SVCD1 PROV_SVCD2	If PROVID_ATT is not blank, then use PROV_SVCD_ATT Else if PROVID1 is not blank, then use PROV_SVCD1 Else if PROVID_ADM is not blank, then use PROV_SVCD_ADM Else if PROVID2 is not blank, then use PROV_SVCD2. Else blank.
Recoded Sponsor Service	CHAR(1)	RSPONSVC	PATCAT_E PATCAT_P	Derived from 1st character of PATCAT. Use PATCAT_E if available. Otherwise, use PATCAT_P. If A, C, F, M, N then retain value. Else if B then assign O. Else if P then assign H. Else if R then assign 4. Else if PATCAT is K71 or K78 then assign 4.

Field	Format	SAS Name	Source Element	Transformation
				Else assign X.
Segment Days 1-8 (Enc History)	N(8)	DAYS_*J*, *J* = 1 to 8	ADM_DT_TM DISCHARGE_DT_TM TRANSACT_DT_TM	See Appendix E.
Segment Duration 1-8 (Enc History)	N(8)	DURATION_*J*, *J* = 1 to 8	ADM_DT_TM DISCHARGE_DT_TM TRANSACT_DT_TM	See Appendix E. Format HHMM8.
Segment Ending Date/Time 1-8 (Enc History)	N(8)	END_TRANS_DT_*J*, *J* = 1 to 8	TRANSACT_DT_TM	See Appendix E. Format e8601dt.
Segment Medical Service 1-8 (Enc History)	CHAR(30)	MEDSVC_*J*, *J* = 1 to 8	MEDICAL_SERVICE_EH	See Appendix E.
Segment MEPR3 Code 1-8 (Enc History)	CHAR(3)	MEPR3_*J*, *J* = 1 to 8	ENCOUNTER HISTORY LOC_NURSE_UNIT_REF LOCATION MEPRS_CD	See Appendix E.
Segment Nurse Location 1-8 (Enc History)	CHAR(25)	NURSELOC_*J*, *J* = 1 to 8	ENCOUNTER HISTORY LOC_NURSE_UNIT_REF LOCATION UNIT_DISPLAY	See Appendix E. Format e8601dt.
Segment Service Line 1-8 (Enc History)	CHAR(5)	SERVICE_LINE_*J*, *J* = 1 to 8	MEPR3_*J*	For *J* = 1 to 30, 1st apply $slmep3a. format from Table 1b to MEPR3. 2nd apply $slmep4a. from Table1b to MEPRS_CD
Segment Starting Date/Time 1-8 (Enc History)	N(8)	START_TRANS_DT_*J*, *J* = 1 to 8	TRANSACT_DT_TM ADM_DT_TM	See Appendix E.
Service Line (Discharge)	CHAR(5)	SERVICE_LINE	MEPRS_CD MEPR3	Apply $slfmt. format from Table 1b to MEPR3.
Skill Type based on Primary HIPAA Taxonomy For Additional Provider 1	CHAR(2)	SKILLH2	HIPAA1	Apply SKILLTYPE&fy.H format from Table 1b to PROV_HIPAA2
Skill Type based on Primary HIPAA Taxonomy For Additional Provider 2	CHAR(2)	SKILLH3	HIPAA1	Apply SKILLTYPE&fy.H format from Table 1b to PROV_HIPAA3
Skill Type based on Primary HIPAA Taxonomy For Additional Provider 3	CHAR(2)	SKILLH4	HIPAA1	Apply SKILLTYPE&fy.H format from Table 1b to PROV_HIPAA4
Skill Type based on Primary HIPAA Taxonomy For Additional Provider 4	CHAR(2)	SKILLH5	HIPAA1	Apply SKILLTYPE&fy.H format from Table 1b to PROV_HIPAA5
Skill Type based on Primary HIPAA Taxonomy For Additional Provider 5	CHAR(2)	SKILLH6	HIPAA1	Apply SKILLTYPE&fy.H format from Table 1b to PROV_HIPAA6
Skill Type based on Primary HIPAA Taxonomy For Admitting Physician	CHAR(2)	SKILLH_ADM	HIPAA1	Apply SKILLTYPE&fy.H format from Table 1b to PROV_HIPAA_ADM
Skill Type based on Primary HIPAA	CHAR(2)	SKILLH1	HIPAA1	Apply SKILLTYPE&fy.H format from Table 1b to PROV_HIPAA1

Field	Format	SAS Name	Source Element	Transformation
Taxonomy For Appointment Provider				
Skill Type based on Primary HIPAA Taxonomy For Attending Physician	CHAR(2)	SKILLH_ATT	HIPAA1	Apply SKILLTYPE&fy.H format from Table 1b to PROV_HIPAA_ATT
Skill Type based on Primary HIPAA Taxonomy For Primary Provider	CHAR(2)	SKILLH_PRIM	PROVID_ADM PROVID_ATT PROVID1 PROVID2 SKILLH_ADM SKILLH_ATT SKILLH1 SKILLH2	If PROVID_ATT is not blank, then use SKILLH_ATT Else if PROVID1 is not blank, then use SKILLH1 Else if PROVID_ADM is not blank, then use SKILLH_ADM Else if PROVID2 is not blank, then use SKILLH2. Else blank.
Skill Type based on Primary HIPAA Taxonomy For Referring Physician	CHAR(2)	SKILLH_REF	HIPAA1	Apply SKILLTYPE&fy.H format from Table 1b to PROV_HIPAA_REF
Sponsor Person ID Type Code	CHAR(1)	SPONSIDTYPE	SPONSSN	S, when SPONSSN is not blank. Otherwise, blank.
Transaction Date & Time (Enc History) (NOT RETAINED)	N(8)	TRANSACT_DT_TM	TRANSACTION_DT_TM TRANSACTION_TM_ZN	Run macro %local_offset using DT_TM & TM_ZN variables for each date to get _LOCAL variables & rename. Apply format e8601dt.
Treatment MEPRS 1 Code	CHAR(1)	MEPR1	MEPRS_CD	The 1st digit of the MEPRS_CD (substr(MEPRS_CD,1,1))
Treatment MEPRS 2 Code	CHAR(2)	MEPR2	MEPRS_CD	The 1st 2 digits of the MEPRS_CD (substr(MEPRS_CD,1,2))
Treatment MEPRS 3 Code	CHAR(3)	MEPR3	MEPRS_CD	The 1st 3 digits of the MEPRS_CD (substr(MEPRS_CD,1,3))
Treatment Parent MTF	CHAR(4)	MTF_PARENT	LOCATION MTF_PARENT_G DMISID INDEX MTF_PARENT_D	Use MTF_PARENT_D if not blank. Otherwise, use MTF_PARENT_G.

VIII. REFRESH FREQUENCY

Weekly

IX. DATA MARTS

X. SPECIAL OUTPUTS

None at this time.

APPENDIX A: Administrative Text Processing Steps and Field Additions

The following processes will be applied to the Encounter file.

1. MPI Merge: See the MPI specification for appending EDIPN, SPONSSN, and PARC.

2. LVM Merge: Append the Enrollment DMISID (MTF_ENR), Alternate Care Value (ACV), Alternate Care Value Group (ACVGROUP), Health Care Delivery Program Code (HCDP), Assigned HCDP (HCDP_ASSGN), Beneficiary Category (BENCAT), Common Beneficiary Category (COMBEN), PCM ID (PCMIDLVM), PCM Type (PCM_TYPE), Medicare Flag (MEDICARE_FLAG), Sponsor Service Aggregate (SAGGLVM), Sponsor Service (SSVCLVM), Patient Privilege Code (PRIVILEGE), Gender (GENDER), Date of Birth (PATDOB_CHAR), Race Code (RACE), Ethnic Group (ETHNIC), Marital Status (MARITAL), Eligibility Group (ELG_GRP), Enrollment Group (ENR_GRP), TPR Eligibility Flag (TPRELIG) and TRICARE Young Adult Flag (TYAFLAG) from the longitudinal LVM for Encounter data. (This merge occurs after the MPI merge described above and occurs on the "whole" Encounter dataset, not just the newly processed records):
   a. Merge to the LVM by EDIPN for the FM of the discharge date, if available, or the admission date, if not.
   b. If a match is found, assign all variables as described in Table 2. (If these values are missing/blank from LVM, then the fields remain missing/blank).

3. DMIS Merge: Merge to the MDR DMIS ID Index based on discharge date, if available, or the admission date, if not, and treatment MTF to append the T17 Region (MTF_T17_REG), T3 Region (MTF_T3_REG), Treatment Service (MTF_SVC), Multi-Service Market Area (MTF_MSMA), and Treatment Major Command (MTF_CMND).

   Merge to the MDR DMIS ID Index based on discharge date, if available or the admission date, if not, and enrollment DMIS ID (MTF_ENR) to append the Enrollment T17 Region (ENR_T17_REG), T3 Region (ENR_T3_REG), Enrollment Parent Site (ENR_PARENT) and Treatment Service (ENR_SVC).

4. CAD Merge: Merge to the MDR CAD based on Patient Zip, Sponsor Service (after mapping to A, F, N and O), and the CAD matching the discharge date, if available or the admission date, if not. (If Patient Zip is not usable, the Treatment MTF Zip Code is used in its place.) The fields Patient Catchment Area (CATCH), Patient T17 Region (BEN_T17_REG), T3 Region (BEN_T3_REG), MTF Service Area (MTFSVCAREA) and PRISM Area MTF (PRISM) are added in this process.

5. MDR DMHRSi Basic HR Merge: Add provider information from the MDR DMHRSi Basic HR file by merging the Provider's DMHRSi extract records for the given discharge date, if available or the admission date, if not, and provider identifier. Note that only DMHRSi extract records that have both a defined start (ASSIG_START) and end (ASSIG_END) date should be considered.

   Since more than one applicable Provider DMHRSi extract record may coincide with the discharge date, if available or the admission date, if not, the following identifiers should be tried, in order, until a definitive match is found, using the sequence of steps below.
   - Provider EDIPN (PROVEDIPN*K*)
   - Provider NPI (PROVNPI*K*)

a. Determine the set of DMHRSi extract records for the provider identifier in which the start and end dates bound the discharge date, if available or the admission date, if not.
b. If that set is empty, then no definitive match is possible.
c. Determine the subset of DMHRSi extract records with the latest processing date (PROCDATE).
d. If the records in that set do not all share identical start and end dates, then no definitive match is possible.
e. If the records in that subset do not yield identical results (ASSIG_DMISID, ORG_UIC, ORG_ID, 1-character mapped value of SERVICE, 1-character mapped value of ASSIG_SERVICE, PERSON_TYPE, and PG_ASSIG_FCC), then no definitive match is possible.
f. If the records in that subset all yield empty or missing results, then no definitive match is possible.
g. Otherwise, a definitive match has been found, use the results to assign all variables, as described in Table 2.

If no definitive match is found using all possible provider identifiers for a given provider, then PROVMTFD*K*, PROVMEPRD*K,* PROVORGD*K*, PROVUICD*K*, and PROVCATD*K* will be set to 'NONE'; PROVSVCD*K* and PROVSVCASSGD*K* will be set to 'Z'.

APPENDIX B: MS-DRG Grouping

Grouping of MDR data for Medicare Severity Diagnosis Related Group (MS-DRG) codes in the MDR Admission file is performed by the 3M Grouper Plus System (GPS), which uses a set of Java classes that are resident on the primary MDR processing node. The subset of GPS capabilities that are necessary for MDR Admission grouping is exposed to the SCE via the Java classes of the MDR Grouping Client (MGC), and the MDR Processing Utilities offer SAS macros for both MDR processing and SCE users to submit grouping requests through the MGC to the GPS.

For SCE users, the SAS include file, scegpsdrg.sas, is provided in the MDR Processing Utilities, located in /mdr/aprod/util. This include file provides the necessary %INCLUDE statements for performing MS-DRG grouping. Among those, gpsdrg.sas provides the key macros for grouping, while the other include files offer a variety of convenience macros.

Only the macro named %gpsdrg_mdr_sidr is used in MDR Admission processing. This macro performs the MS-DRG grouping on the MDR Admission data. In general, this macro requires the name of an appropriate data set. On each observation in that data set, the input fields for grouping are duplicated, transformed, and submitted to the MGC and the results of that submission are parsed, transformed, and assigned to the grouping output fields. For efficiency, specific observations that cannot be grouped, such as those for which no procedure codes are defined, have been removed as much as possible from the MDR Admission data set prior to grouping.

MS-DRG GROUPING OF THE MDR ADMISSION FILE

Grouping of MDR Admission data for MS-DRG codes is accomplished during MDR processing via the %gpsdrg_mdr_sidr() macro. The method for grouping MDR Admission data can be discerned by following the logic of the macro, which is presented below.

1. All MDR Admission observations have a disposition date in fiscal year 2017 or later so they will always be MS-DRG grouped with Hospital Acquired Condition (HAC)/Present on Admission (POA) processing enabled, if possible. For GENESIS data, the POA variable in the Bulk Data Extract (BDE) is not populated so the HAC/POA process has been turned off until that variable starts being processed.
2. For each observation in the file:
    a. The type of ICD encoding (i.e., 9 or 10) is determined based on the disposition date or end date of care. (Note: the MDR Admission file will only have ICD-10 encoding.)
    b. The desired version and fiscal year of the grouper to execute is determined based on the disposition date (datepart(DISCHARGE_DT_TM)).
    c. Diagnosis codes (DX_CCE1-DX_CCE20), POA indicators, and procedure codes (PROC_1-PROC_20) are formatted for grouping. At this time all POA indicators are blank for the MDR Admission file.
    d. Gender (PATSEX) and age (PATAGE) are assigned and formatted for grouping.
    e. Dates are formatted for grouping. Dates include admission date (ADM_DT_TM), discharge date (DISCHARGE_DT_TM) and patient date of birth (PATDOB).
    f. In the future, the HAC/POA processing flag will be assigned during this step.
    g. The disposition code (DISPCODE_LEGACY) is identified.
    h. The observation is submitted to the MGC.

i. Output values are retrieved from the return string.
j. A frequency of grouper return codes and versions is produced on the listing.

3. Output fields are assigned their values as the three lots are merged back into the data set. Output fields include:
   a. MSDRG – the MSDRG code returned from the grouper for each observation
   b. MSDRGVRSN – the version of the MSDRG grouper used to group each observation
   c. MSMDC – the MDC code returned from the grouper for each observation
   d. MSDRGRTC – the return code of each observation from the grouper

SAS CODE TO CREATE INPUT FILE & MERGE DATA BACK TO FINAL FILE

```
data grouper (drop = adm_dt_tm proc_1-proc_20);
      informat dispdate admdate birtdate date9.;
            set admission (keep = gen_msdrg dispcode_legacy Discharge_DT_TM
            encounter_nk adm_infr_flag patdob patsex gen_mdc dx_cce1-dx_cce20
            proc_1-proc_20 adm_dt_tm);
      if adm_infr_flag = 0;

      if (Discharge_DT_TM = .) then delete;
      if (adm_dt_tm = .) then delete;
      if dx_cce1 = '' then delete;

      dispdate= datepart(Discharge_DT_TM);
      admdate = datepart(adm_DT_TM);
      birtdate = input(patdob,yymmdd8.);

      * name variables appropriately for grouper *;
      rename      patsex = dmissex
                  dispcode_legacy = recdisp
                  ;

      format MSDRG $3. MSMDC $2. MSDRGRTC z2. MSDRGVRSN $4.;

      MSDRG         = '';
      MSMDC    = '';
      MSDRGRTC  = '';
      MSDRGVRSN = '';

      n_dischdt = dispdate;

      * rename proc_1 to proc1 *;
      array xpx[20] $7. proc1-proc20;
      array xproc[20] $7. proc_1-proc_20;

      * add poa variable based on DX - use N for now *;
      array xdx[20] $7. dx1-dx20;
      array xdiag[20] $7. dx_cce1-dx_cce20;
      array xpoa[20] $1. DX1POA DX2POA DX3POA DX4POA DX5POA
                  DX6POA DX7POA DX8POA DX9POA DX10POA
                  DX11POA DX12POA DX13POA DX14POA DX15POA
                  DX16POA DX17POA DX18POA DX19POA DX20POA;

      do j = 1 to 20;

            * rename proc variable *;
```

```
            xpx[j]  = xproc[j];

            * rename diag variable *;
            xdx[j]  = xdiag[j];

            * assign POA based on DX *;
         if xdx[j] = '' then xpoa[j]='';
            else xpoa[j] = '';

      end;
      drop j;
run;
*
.
;
%include "/mdr/genesis/aprod/admission/scegpsdrg_admission.sas";
%gpsdrg_mdr_sidr(grouper)

data grouper;
 set grouper;
 version = substr(msdrgvrsn,1,2);
 MTF_DAYS = dispdate - admdate;
run;

proc sql;
create table admission as
 select a.*,
          g.msdrg, g.msmdc, g.msdrgvrsn, g.msdrgrtc, g.version, g.mtf_days, g.DX1POA, g.DX2POA,
         g.DX3POA, g.DX4POA, g.DX5POA, g.DX6POA, g.DX7POA, g.DX8POA, g.DX9POA, g.DX10POA,
         g.DX11POA, g.DX12POA, g.DX13POA, g.DX14POA, g.DX15POA, g.DX16POA,g.DX17POA,
         g.DX18POA, g.DX19POA, g.DX20POA
 from admission as a left join grouper as g
 on   (a.ENCOUNTER_NK = g.ENCOUNTER_NK)
 ;
quit;
```

Appendix C. MS-DRG Relative Weighted Product (MSDRGRWP) Algorithm FY17 and Forward

Direct Care Inpatient:

MSDRGBASERWP = MS-DRG Baseline RWP
MSDRGOUTRWP = Plus/Minus RWP due to long/short stay outlier
MSDRGRWP = Total RWP, summation of MSDRGBASERWP and MSDRGOUTRWP

From the TRICARE MS-DRG Weights file (FY specific until 12/31/2019, CY specific for CY20 and forward):
LST = Long Stay Threshold
SST = Short Stay Threshold
GMLOS = Geometric Mean Length of Stay
WEIGHT = MS-DRG weight
PERDIEM = Weight / GMLOS

1. Merge the Admission file by the MSDRG with the appropriate TRICARE MS-DRG weights file (see Table 1b).
    a. For FY18 and before, use the TRICARE MS-DRG weights based on the appropriate FY.
    b. For FY19 & CY19, use the FY19 weights file.
    c. For CY20 and later, use the weights file of the appropriate CY.

2. Define each Admission record into the following categories (MSDRGICAT):

    MSDRGICAT = 2 (Direct In, Transfer Out): When disposition status is discharged to a short-term facility (RECDISP = 02 or 2) and admission source (ADMIT_SOURCE) is Born Inside the Hospital, Clinic Referral, Court/Law Enforcement, Non-Health Care Facility Source of Origin, Patient/Self, Born Outside this Hospital, Transfer from Hospital Inpt - New Claim.

    MSDRGICAT = 3 (Transfer In, Transfer Out): When disposition status is discharged to a short-term facility (RECDISP = 02 or 2) and admission source (ADMIT_SOURCE) is Transfer from a Civilian/Community Hospital, Transfer from Ambulatory Surgery Center, Transfer from another HC Facility, Transfer from SNF.

    MSDRGICAT = 4 (Transfer In, Direct Out): When disposition status is NOT discharged to a short-term facility (RECDISP not equal to 02 or 2) and admission source (ADMIT_SOURCE) is Transfer from a Civilian/Community Hospital, Transfer from Ambulatory Surgery Center, Transfer from another HC Facility, Transfer from SNF.

    Otherwise MSDRGICAT = 1 (Direct In, Direct Out)

3. Create a bed day field (CALCDAYS) setting value equal to MTF_DAYS, which is the discharge date (datepart(DISCHARGE_DT_TM)) less the admission date (datepart(ADM_DT_TM)). If MTF_DAYS equal to zero (0), then set CALCDAYS equal to 1.

4. For MSDRGs 998 and 999 (ungroupable) set MSDRGBASERWP and MSDRGOUTRWP equal to 0 (zero).

5. For MSDRGs 610, 611, 613, 632, and 635 assign MSDRGBASERWP and MSDRGOUTRWP as follows:

If CALCDAYS <= LST then
MSDRGBASERWP = Weight
MSDRGOUTRWP = 0

If CALCDAYS > LST then
MSDRGBASERWP = Weight
MSDRGOUTRWP = PERDIEM*0.33*(CALCDAYS – LST)

6. For MSDRGICAT = 1, 3, or 4

If SST < CALCDAYS <= LST then
MSDRGBASERWP = Weight
MSDRGOUTRWP = 0

If CALCDAYS <= SST then
MSDRGBASERWP = Weight
MSDRGOUTRWP = Min(Weight, PERDIEM*2*CALCDAYS) – Weight
(Note: This will produce either a negative value or zero.)

If CALCDAYS > LST then
MSDRGBASERWP = Weight
MSDRGOUTRWP = PERDIEM*0.33*(CALCDAYS – LST)

7. For MSDRGICAT = 2

For MSDRGs 612, 631, 633, 634, 636, 646-651, 676-681, 787-794:

If CALCDAYS > LST then
MSDRGBASERWP = Weight
MSDRGOUTRWP = PERDIEM*0.33*(CALCDAYS – LST)

If CALCDAYS <= LST then
MSDRGBASERWP= Weight
MSDRGOUTRWP = (Min(Weight, (2*PERDIEM)+((1.25*PERDIEM)*(CALCDAYS-1)))) - Weight

For all other MSDRGs:

If CALCDAYS > LST then
MSDRGBASERWP = Weight
MSDRGOUTRWP = PERDIEM*0.33*(CALCDAYS – LST)

If CALCDAYS <= LST then
MSDRGBASERWP = Weight
MSDRGOUTRWP = (Min(Weight, (2*PERDIEM)+(PERDIEM*(CALCDAYS-1)))) - Weight

Appendix D. Derivation for Specified Variables

Table D1: Diagnosis Group Derivation

ICD-10 First 3 digits (FY16 and forward)	Category Number	Disease Category Name
A00-B99	1	Infections & Parasites
C00-D49	2	Neoplasms
E00-E89	3	Endocrine & Metabolism
D50-D89	4	Blood
F01-F99	5	Mental
G00-H95	6	Nerves and Senses
I00-I99	7	Circulatory System
J00-J99	8	Respiratory System
K00-K95	9	Digestive System
N00-N99	10	Genitourinary
O00-O9A	11	Pregnancy and Childbirth
L00-L99	12	Skin
M00-M99	13	Musculoskeletal
Q00-Q99	14	Congenital Anomalies
P00-P96	15	Perinatal
R00-R99	16	Ill-Defined
S00-T88	17	Injury & Poisoning
Z00-Z99	18	Supplementary Classifications
V00-Y99	19	Unknown (external causes)
Anything starting with "DOD"	20	DOD unique codes
All Others		blank

Appendix E. Derivation for Clinical Segment Variables

1. Pull data from the Encounter History file (enco_cehistory) for each ENCOUNTER_NK in the Encounter file (enco_cencounter) where the Encounter Type = 'Inpatient' (apply $patient_type. format to PATIENT_TYPE_REF to get Encounter Type). Keep the following variables from the Encounter History and Encounter files:

   Encounter History
   - ENCOUNTER_NK
   - ENC_HISTORY_SK
   - ENCNTR_TYPE_REF
   - LOC_NURSE_UNIT_REF
   - MEDICAL_SERVICE_REF
   - TRANSACTION_DT_TM
   - TRANSACTION_TM_ZN

   Encounter
   - ADM_DT_TM
   - DISCHARGE_DT_TM
   - PATDOB

2. Merge to Location file (mdr/pub/ genesis/location) by matching LOCATION_SK TO LOC_NURSE_UNIT_REF to get the UNIT_DISPLAY and MEPRS_CD variables.

   Location
   - UNIT_DISPLAY
   - MEPRS_CD

3. Prepare the data set for identifying clinical segments, including:
   a. Calculating the correct time zone. See TRANSACT_DT_TM.
   b. Apply formats and modify lengths of variables as follows:
      i. MEPR3_EH = SUBSTR(MEPRS_CD,1,3)
      ii. MEDICAL_SERVICE_EH = PUT(MEDICAL_SERVICE_REF, $MEDICAL_SERVICE.)
      iii. ENCOUNTER_TYPE_EH = PUT(ENCNTR_TYPE_REF, $PATIENT_TYPE.)

4. Create the PREADMIT_FLAG, FOLLOW_UP_FLAG and DURING_STAY_FLAG.
   a. PREADMIT_FLAG is based on the TRANSACT_DT_TM, ADM_DT_TM & the ENCOUNTER_TYPE_EH variables as follows:
      i. If the TRANSACT_DT_TM >= ADM_DT_TM or the ENCOUNTER_TYPE_EH = Inpatient then the PREADMIT_FLAG = 0
      ii. Else if the TRANSACT_DT_TM < ADM_DT_TM then the PREADMIT_FLAG is defined as follows:

TABLE E1: PREADMIT_FLAG Derivation

ENCOUNTER_TYPE_EH	PREADMIT_FLAG
Preadmit	1
Prereg	2
Emergency	3
Outpatient	4
Outpatient in a Bed	5
Outpatient Hold	6
Outpatient Day Surgery	7
Observation	8
Referral Tracking	9
Clinic	10
Dental	11
PreRecurring	12
Between Visit	13
(All Other Values)	99

b. FOLLOW_UP_FLAG is based on the TRANSACT_DT_TM, DISCHARGE_DT_TM and ENCOUNTER_TYPE_EH as follows:
    i. If the TRANSACT_DT_TM <= DISCHARGE_DT_TM or the ENCOUNTER_TYPE_EH = Inpatient then the FOLLOW_UP_FLAG = 0
    ii. Else if the TRANSACT_DT_TM > DISCHARGE_DT_TM then the FOLLOW_UP_FLAG = 1
c. DURING_STAY_FLAG is based on the TRANSACT_DT_TM, ADM_DT_TM, DISCHARGE_DT_TM and ENCOUNTER_TYPE_EH as follows:
    i. If the TRANSACT_DT_TM < ADM_DT_TM or the TRANSACT_DT_TM > DISCHARGE_DT_TM or the ENCOUNTER_TYPE_EH = Inpatient then the DURING_STAY_FLAG = 0
    ii. Else if the ADM_DT_TM <= TRANSACT_DT_TM <= DISCHARGE_DT_TM then the DURING_STAY_FLAG = 1

5. Identify the start and stop dates and times of up to 30 Clinical Segments based on MEPRS code, Location, Medical Service and calculate the number of days the patient stayed in each segment (based on the SIDR logic) and the actual duration of the patient's stay in hours and minutes. See Table E2 below for the output variable names and formats.
    a. For the 1st segment the start date is the ADM_DT_TM. The end date is the TRANSACT_DT_TM when the MEPR3_EH, UNIT_DISPLAY or MEDICAL_SERVICE_EH changes.
    b. The start date of the 2nd – 29th (or 2nd to last) segment is the end date of the previous segment. The end date of these segments is the TRANSACT_DT_TM when the MEPR3_EH, UNIT_DISPLAY or MEDICAL_SERVICE_EH changes.
    c. For the last (or 30th segment), the start date is the end date of the previous segment. The end date of these segments is the DISCHARGE_DT_TM.

d. If the ENCOUNTER_TYPE_EH is 'Preadmit,' 'Prereg,' 'Between Visit' or 'Referral Tracking' then make the start and end dates for that segment the same.

Table E2: Non-Duration Variables in Clinical Segments

Field	Format	SAS Name	Source Element	Transformation
Encounter Type 1-8 (Enc History)	CHAR(30)	ENCTYPE_*J*, *J* = 1 to 8	ENCOUNTER_TYPE_EH	For each new clinical segment [n], ENCTYPE_[n] = ENCOUNTER_TYPE_EH for the START_TRANS_DT_[n] = TRANSACT_DT_TM record in the Encounter History file. Keep the 1st 8 of up to 30 segments.
Ending Transaction Date/Time 1-8 (Enc History)	N(8)	ENC_TRANS_DT_*J*, *J* = 1 to 8	TRANSACT_DT_TM DISCHARGE_DT_TM	Format as e8601dt. Keep the 1st 8 of up to 30 segments.
Medical Service 1-8 (Enc History)	CHAR(30)	MEDSVC_*J*, *J* = 1 to 8	MEDICAL_SERVICE_EH	For each new clinical segment [n], MEDSVC_[n] = MEDICAL_SERVICE_EH for the START_TRANS_DT_[n] = TRANSACT_DT_TM record in the Encounter History file. Keep the 1st 8 of up to 30 segments.
MEPRS Code 1-8 (Enc History)	CHAR(3)	MEPR3_*J*, *J* = 1 to 8	MEPRS_EH	For each new clinical segment [n], MEPR3_[n] = MEPRS_EH for the START_TRANS_DT_[n] = TRANSACT_DT_TM record in the Encounter History file. Keep the 1st 8 of up to 30 segments.
Nurse Location 1-8 (Enc History)	CHAR(25)	NURSELOC_*J*, *J* = 1 to 8	UNIT_DISPLAY	For each new clinical segment [n], NURSELOC_[n] = UNIT_DISPLAY for the START_TRANS_DT_[n] = TRANSACT_DT_TM record in the Encounter History file. Keep the 1st 8 of up to 30 segments.
Starting Transaction Date/Time 1-8 (Enc History)	N(8)	START_TRANS_DT_*J*, *J* = 1 to 8	TRANSACT_DT_TM	Format as e8601dt. Keep the 1st 8 of up to 30 segments.

6. Calculate the days and duration variables for each segment. See Table E3 for the output variable names and formats.

Table E3: Duration Variables in or Based on Clinical Segments

Field	Format	SAS Name	Source Element	Transformation
Duration 1-8 (Enc History)	HHMM8.	DURATION_*J*, *J* = 1 to 8	END_TRANS_DT_1 – END_TRANS_DT_8 START_TRANS_DT_1 – START_TRANS_DT_8	INTCK('HOURS',START_TRANS_DT_TM_[I],END_TRANS_DT_[I]) Format as HHMM5. Keep the 1st 8 of up to 30 segments.
Days in Segment 1 – 8 (Enc History)	N(8)	DAYS_*J*, *J* = 1 to 8	END_TRANS_DT_1 – END_TRANS_DT_8 START_TRANS_DT_1 – START_TRANS_DT_8	INTCK('DTDAY', DATEPART(START_TRANS_DT_TM_[I]), DATEPART(END_TRANS_DT_[I])) Keep the 1st 8 of up to 30 segments.
Bed Days from Segments (Enc History)	N(8)	BED_DAYS	DAYS_1 – DAYS_30	Sum of DAYS_1 to DAYS_30.
ICU Days (Enc History)	N(8)	ICU_DAYS	DAYS_1 – DAYS_30 NURSELOC_1-NURSELOC_30	Sum of DAYS_[*J*] when index(upcase(NURSELOC_[*J*]),'ICU')
Bassinet Days (Enc History)	N(8)	BASS_DAYS	DAYS_1 – DAYS_30 NURSELOC_1-NURSELOC_30 PATDOB ADM_DT_TM MEDSVC_1-MEDSVC_30	Sum of DAYS_[I] when 1. index(upcase(NURSELOC_[I]),'LDRP') or index(upcase(NURSELOC_[I]),'LDR') and age <= 120 days or 2. index(upcase(NURSELOC_[I]),'NICU') or 3. DATEPART(ADM_DT_TM) – PATDOB <= 120 days or 4. medsvc_[I] = 'Newborn'

Appendix F. Derivation for Birth Weight

Birthweight is derived from two main sources, the Clinical Events file and the Maternity files. Tables F.1, F.2b and F.2b & F show the variables needed to determine the birthweight, if any, associated with an admission record using these two sets of files.

Clinical Events

The primary source of the birth weight variable is the Clinical Events file.

1. Select all records from the Clinical Events file where the EVENT_CODE_REF is 712070 (Birth Weight), 29335541 (Birth Weight (g):) or 17022176 (Birth Weight:), the record has the latest update date and time (UPDT_DT_TM) on the Clinical Events file, the record matches to a record on the Admission file as currently processed via Encounter_NK and the patient age (PATAGE) is 0.

TABLE F.1: Fields Required from the Clinical Events File

Field	Format	SAS Name	Source Element	Transformation
Encounter Key (Primary)	CHAR(100)	ENCOUNTER_NK	ENCOUNTER_NK	No transformation.
Event Code	CHAR(40)	EVENT_CODE_REF	EVENT_CODE_REF	No transformation.
Result Units	CHAR(40)	RESULT_UNITS_REF	RESULT_UNITS_REF	No transformation.
Result Value	CHAR(255)	RESULT_VALUE	RESULT_VALUE	No transformation.
Update Date/Time	N(8)	UPDT_DT_TM	UPDT_DT_TM	No transformation.

2. If a record matches to multiple EVENT_CODE_REF values then prioritize as follows: 712070 or the latest update time of any other EVENT_CODE_REF.

3. Calculate BIRTH_WEIGHT_C as:

    a. If RESULT_UNITS_REF=271 then BIRTH_WEIGHT_C = 1000 * result_value
    b. If RESULT_UNITS_REF=263 then BIRTH_WEIGHT_C = result_value

Maternity Files

The second source of the birth weight variable are files from the Maternity & Encounter subsets.

1. Identify records of women who gave birth from the Maternity Instance (ld.mate_cpinst) and Child (ld.mate_cpchild) files, excluding ongoing and historical pregnancies and keeping only the most current record from the Instance file. Pull Weight Amount and Unit from the Child file.

TABLE F.2a: Fields Required from Maternity Instance & Child Files

Field	Format	SAS Name	Source Element	Transformation
Pregnancy Instance (mate_cpinst)				
Person ID	CHAR(100)	PERSON_SK	PERSON_SK	No transformation.
Pregnancy Child (mate_cpchild)				
Pregnancy Child File Key	CHAR(100)	PERSON_SK	PERSON_SK	No transformation.
Update Date/Time	N(8)	UPDT_DT_TM	UPDT_DT_TM	No transformation.
Weight Amount	N(8)	WEIGHT_AMT	WEIGHT_AMT	No transformation.
Weight Unit	CHAR(100)	WEIGHT_UNIT_REF	WEIGHT_UNIT_REF	No transformation.

2. Get all Encounter_NKs for the mother from the Admission file as currently processed.

3. Get the child's Encounter_NKs (Related_Encounter_NK) from the Encounter/Encounter Relationship File based on the Encounter_NK(s) of the mother.

TABLE F.2b: Fields Required from Encounter/Encounter Relationship File

Field	Format	SAS Name	Source Element	Transformation
Encounter/Encounter Relationship (enco_ceerltn)				
Encounter Key (Primary)	CHAR(100)	ENCOUNTER_NK	ENCOUNTER_NK	No transformation.
Related Encounter Key (Primary)	CHAR(100)	RELATED_ENCOUNTER_NK	RELATED_ENCOUNTER_NK	No transformation.
Update Date/Time	N(8)	UPDT_DT_TM	UPDT_DT_TM	No transformation.

4. Calculate BIRTH_WEIGHT_M as:

    a. If WEIGHT_UNIT_REF=271 then BIRTH_WEIGHT_M = 1000 * weight_amt
    b. If WEIGHT_UNIT_REF=263 then BIRTH_WEIGHT_M = weight_amt

Combine Clinical Events & Maternity Birth Weights

Combine the Clinical Events-based (BIRTH_WEIGHT_C) and the Maternity-based (BIRTH_WEIGHT_M) using Encounter_NK to create the two variables in the final MHS GENESIS Admission file shown in Table F.3.

1. Calculate BIRTH_WEIGHT as:

    a. If BIRTH_WEIGHT_C is not missing then BIRTH_WEIGHT = BIRTH_WEIGHT_C
    b. Else if BIRTH_WEIGHT_M is not zero (0) then BIRTH_WEIGHT = BIRTH_WEIGHT_M
    c. Else BIRTH_WEIGHT = . (missing).

2. Calculate BW_SOURCE as:

    a. If the Encounter_NK is in both the Clinical Events-based and Maternity-based files then BW_SOURCE = "Both CE & Maternity"
    b. If the Encounter_NK is only in the Clinical Events-based file then BW_SOURCE = "Clin Events"
    c. If the Encounter_NK is only in the Maternity-based file then BW_SOURCE = "Maternity".

TABLE F.3: Fields Sent to MHS GENESIS Admission File

Field	Format	SAS Name	Source Element
Birth Weight	N(8)	BIRTH_WEIGHT	BIRTH_WEIGHT_C BIRTH_WEIGHT_M
Source of Birth Weight Variable	N(8)	BW_SOURCE	ENCOUNTER_NK

Appendix G. AHRQ Low Birth Weight Quality Indicator

The Low Birth Weight Quality Indicator is similar to the other AHRQ Quality Indicators, but its derivation is a bit more complex. The Low Birth Weight variable (albw) has four values:

0 = Non-newborn
1 = Low birth weight newborn
2 = Non-low birth weight newborn
N = Date of Birth missing, not possible to calculate

Use the following flow chart to assign the correct values based on the age, admission source, and diagnosis codes.