24 April 2026

MHS GENESIS Admission BDE 3.0 File for the MDR – Future Specification

Version 1.03.00

MHS GENESIS Admission BDE 3.0 File for the MDR – Future Specification – Version 1.02.00

DHAINC06099402

Revision History

Version	Date	Originator	Para/Tbl/Fig	Description of Change
1.00.00	07/29/2024	D Juckett, K. Hofmann	• Initial document	• Initial document
1.00.01	09/27/2024	D Juckett, K. Hofmann	• Table 3	• Update length Medical Service 1-20
1.00.02	10/17/2024	D Juckett, K. Hofmann	• Appendix H	• Added Appendix H: CCE Fix
1.01.00	01/14/2025	D. Juckett	• Table 3 • Appendix A • Appendix B • Section III • Section IV	• Add T5 Region fields • Add Present on Admission Indicators • Modify logic to add T5 Region • Update MS-DRG Grouping logic • Updated transmission frequency • Modified language for organization
1.02.00	03/23/2026	K. Hutchinson	• Appendix B	• Added wording for handling negative age in days for grouping
1.03.00	04/24/2026	K. Hofmann	• Section V	• Removed the filter that drops test locations. This allows historical data from Ft. Leonard Wood to flow into the data.

I. Background

This specification describes the transformation process required to create the Military Health System (MHS) Data Repository (MDR) GENESIS Encounter table based on data received from the Oracle Bulk Data Extract (BDE) 3.0 feeds.

II. Source

The source system is the MHS GENESIS Oracle Millennium database.

III. Transmission (Format and Frequency)

- Current FY: Weekly.
- Frequency of updates:
  1. Prior FY: Weekly for one quarter (October, November, and December) then semiannually (April, October).
  2. All years prior to prior FY: Annually (October) or on an as needed basis when data corrections or updates are required.

IV. Organization

Output products: FY-based SAS datasets containing all MHS GENESIS Admission records where the discharge date is in the fiscal year. Admission tables are stored as fiscal year datasets located at: /mdr/pub/genesis3/admission/fy[##].sas7bdat.

V. Raw

All records in the MDR MHS Admission table are based on data pulled from the raw MIP Redshift genesis_vw.encounter view. To increase the utility of the MDR MHS Admission table, variables from other MDR tables in the following subject areas have been added: Person, Personnel, and Location. Additionally, variables from the DEERS LVM, Omni CAD, DMHRSi, and DMISID Index tables have been added. Table 1 provides a listing of raw Redshift views that are used during the creation of the MDR Admission table.

Table 1. Date File Merges for MSH MHS GENESIS Admission File

Raw Source Table	View Name	Feed Description
Encounter	genesis_vw.encounter	Contains information related to the encounter including patient, provider, time, and location details.
Encounter Alias	genesis_vw.encntr_alias	Used to look up encounter identifiers such as the FIN.
Encounter Personnel Relationship	genesis_vw.encntr_prsnl_reltn	Contains personnel and their relationship (i.e., admitting physician, attending, etc.) to the encounter.

Raw Source Table	View Name	Feed Description
Schedule Appointment	genesis_vw.sch_appt	Contains appointment and scheduling information related to the encounter record.
Schedule Event Actions	genesis_vw.sch_event_action	Contains additional appointment and scheduling information related to the encounter record.
Person	genesis_vw.person	Contains person level demographic information (i.e, race).
Person Information	genesis_vw.person_info	Contains additional person level information (i.e., patcat).
Encounter Information	genesis_vw.encntr_info	Contains additional encounter level information.
Diagnosis	genesis_vw.diagnosis	Contains ICD-10-Dx diagnosis codes as assigned by the provider and coder.
Clinical Event	genesis_vw.clinical_event	Contains discrete events related to the medical record, including results, medications, documents, and many more.
Personnel	genesis_vw.prsnl	Contains information for all people identified within GENESIS as personnel (i.e., doctors, nurses).
Organization	genesis_vw.organization	Used to look up billing organization details (i.e., Coast Guard, DHP).
Personnel Alias	genesis_vw.prsnl_alias	Used to look up personnel identifiers such as the NPI.
Person Alias	genesis_vw.person_alias	Used to look up key person identifiers such as the EDIPI.
Code Value	genesis_vw.code_value	A reference table used to look up descriptions for many code values.
Billing Entity	genesis_vw.billing_entity	Contains descriptions of billing entities.
Health Plan	genesis_vw.health_plan	Contains descriptions of health plans such as TRICARE PRIME.
Time Zone	mdr_processing.vw_gen_mdr_time_zone	Used to identify the MTF time zone to convert UTC datetimes to local time.

Raw Source Table	View Name	Feed Description
Test Patients	mdr_processing.vw_gen_test_pats	Used to remove test patients from the final dataset.
Encounter Health Plan Relationship	genesis_vw.encntr_plan_reltn	Used to look up Patient Identification Process (PIP) fields
Person Organization Relationship	genesis_vw.person_org_reltn	Used to look up Patient Identification Process (PIP) fields
Health Plan	genesis_vw.health_plan	Used to look up Patient Identification Process (PIP) fields

Table 2 contains the descriptions of the MDR tables that are merged to the admission data that allows for additional fields to be added. It includes the merge key (e.g., PERSON_SK).

Table 2. Format File Merges for MDR MHS GENESIS Admission File

Merge	Date Matching	Additional Matching Methodology	Purpose
MDR GENESIS Person Table	N/A	PERSON_SK	Adds patient-related fields from GENESIS such as the patient's name and other demographic fields (i.e., race, ethnicity)
MDR GENESIS Personnel Table	N/A	PERSON_SK	Adds provider-related fields from GENESIS such as the attending physician's name.
MDR GENESIS Location Table	N/A	CURRENT_LOC	Adds the MTF, MEPRS Code, time zone and other location related fields for the encounter.
DEERS Longitudinal VM (LVM)	Visit date between the begin and end dates associated with the segment	EDIPN, SPONSSN	Adds patient-related fields from DEERS such as enrollment, gender, race, date of birth, etc.
Omni CAD	Visit Date	PATZIP Sponsor Service	Adds patient-related geographic concepts such as catchment, PRISM, beneficiary region, etc.
DMHRSi	Visit Date	Provider EDIPN, Provider NPI	Adds provider-related DHA fields such as assigned MTF, provider military service, etc.
DMISID Index	Visit Date	MTF	Adds DMIS ID-related fields such as branch of service, network, TRICARE region, etc.

Upon matching to the MDR MHS GENESIS Person file, MDR MHS GENESIS Personnel file, and MDR MHS 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.

VI. File Layout

The MDR GENESIS Admission file is stored in a SAS data set. Table 3 provides the file layout and processing rules.

Table 3. Fields in the MDR MHS GENESIS Admission Table

Field	Format	SAS Name	Source Element	Transformation Rule
Encounter				
Admission Date	Num(8)	ADM_DT	encounter.admit_dt_tm	datepart(admit_dt_tm) formatted as YYMMDDN
Admission Date and Time	Num(8)	ADM_DT_TM	encounter.admit_dt_tm	
Admission Location (Composite Description)	Char(45)	ADMISSION_LOC_COMPOSITE	encounter.location_cd	Return the unit_display
Admission Date & Time	Num(8)	ADMIT_DT_TM	encounter.admit_dt_tm	No Transformation
Admission Mode	Char(60)	ADMIT_MODE	encounter.admit_mode_cd	code_set = 68 and active_ind = 1
Admission Source	Char(43)	ADMIT_SOURCE	encounter.admit_src_cd	code_set = 2 and active_ind = 1
Admission Source Legacy	Char(1)	ADMIT_SOURCE_LEGACY	encounter.admit_src_cd	Join to code_value table where admit_src_cd matches the code_value and code_set = 2 and active_ind = 1 and retrieve display.
Admission Type	Char(32)	ADMIT_TYPE	encounter.admit_type_cd	Join to code_value table where admit_type_cd matches the code_value and code_set = 3 and active_ind = 1 and retrieve display.
Arrival Date & Time	Num(8)	ARRIVE_DT_TM	encounter.arrive_dt_tm	Converted to local time.
Discharge Location	Char(40)	CURRENT_LOC	encounter.location_cd	No Transformation
Departure Date & Time	Num(8)	DEPART_DT_TM	encounter.disch_dt_tm	Converted to local time.
Discharge Date	Num(8)	DISCHARGE_DT	encounter.disch_dt_tm	datepart(disch_dt_tm) formatted as YYMMDDN
Discharge Date & Time	Num(8)	DISCHARGE_DT_TM	encounter.disch_dt_tm	Converted to local time.

Field	Format	SAS Name	Source Element	Transformation Rule
Discharge Disposition Code	Char(60)	DISPCODE	encounter.disch_disposition_cd	Join to code_value table where discharge_disposition_cd matches the code_value and code_set = 19 and active_ind = 1 and retrieve display.
Disposition Code, Legacy	Char(2)	DISPCODE_LEGACY	encounter.disch_disposition_cd	if dispcode in ('Home or Self Care' 'Advice Assessment' 'Released Without Limitations' 'Returned to Duty' 'Sick at Home/Quarters' 'RR - Results Received' 'Released With Work Duty Limitations') then dispcode_legacy = '01'; else 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') then dispcode_legacy = '02'; else if dispcode in ('Transfer to SNF') then dispcode_legacy = '03'; else if dispcode in ('Against Medical Advise' 'Elopement' 'Left Without Being Seen') then dispcode_legacy = '07'; else if dispcode in ('Patient has expired') then dispcode_legacy = '20'; else if dispcode in ('Admitted as inpatient' 'Continued Stay' 'Transfer to Another Clinical Service') then dispcode_legacy = '30'; else if dispcode in ('Discharged to Other Federal Facility') then dispcode_legacy = '43'; else if dispcode in ('Discharged to ICF') then dispcode_legacy = '63'; else if dispcode in ('Transfer to Other') then dispcode_legacy = '70'; else if dispcode in ('Place in Observation' 'Referred to ER') then dispcode_legacy = '72'; else dispcode_legacy = '';
DEERS Patient Identifier (EDIPN)	Char(10)	EDIPN	encounter.person_id	Join to person_alias where person_id matches and person_alias_type_cd = 22 and active_ind = 1
Encounter Key (Secondary)	Num(8)	ENCNTR_ID	encounter.encntr_id	No transformation.

Field	Format	SAS Name	Source Element	Transformation Rule
Encounter Class	Char(33)	ENCOUNTER_CLASS	encounter.encntr_type_class_cd	Join to code_value table where encntr_type_class_cd matches the code_value and code_set = 69 and active_ind = 1 and retrieve display.
Encounter Key (Secondary)	Char(100)	ENCOUNTER_SK	encounter.encntr_id	Derive as character version of encntr_id using put function: encounter_sk = put(encntr_id, 21. -L);
Encounter Status	Char(25)	ENCOUNTER_STATUS	encounter.encntr_status_cd	Join to code_value table where encntr_status_cd matches the code_value and code_set = 261 and active_ind = 1 and retrieve display.
Encounter Type	Char(26)	ENCOUNTER_TYPE	encounter.encntr_type_cd	Join to code_value table where encntr_type_cd matches the code_value and code_set = 71 and active_ind = 1 and retrieve display.
MDR Original Encounter Extract Date	Char(8)	EXTRDATE_O	Encounter.dl_eff_dt	Character format of dl_eff_dt.
GENESIS Extract Date	Num(8)	EXTRDATE_UTC	Encounter.dl_eff_dt	Apply format e8601dt to dl_eff_dt.
Financial Information Number (FIN)	Char(40)	FIN	encounter.encntr_id	Join to encntr_alias table where encntr_id matches and encntr_alias_type_cd = 1077 and active_ind = 1 and end_effective_dt_tm > sysdate and retrieve alias.
Financial Classification (Payment Source)	Char(58)	FINANCIAL_CLASS	encounter.financial_class_cd	Join to code_value table where financial_class_cd matches the code_value and code_set = 354 and active_ind -1 and retrieve display.
Inpatient Admit Date & Time	Num(8)	INPATIENT_ADMIT_DT_TM	encounter.inpatient_admit_dt_tm	Converted to local time.
Admission Location	Num(8)	LOCATION_CD	encounter.location_cd	Join to MDR Location table where the location_cd matches and return loc_nurse_unit_disp.
Medical Service (Encounter)	Char(43)	MEDICAL_SVC	encounter.med_service_cd	Join to code_value table where med_service_cd matches the code_value and code_set = 34 and active_ind = 1 and retrieve display.
Treatment MEPRS Code	Char(4)	MEPRS_CD	encounter.location_cd	Join to MDR Location table where location_cd matches and return MEPRS field.

Field	Format	SAS Name	Source Element	Transformation Rule
Military Treatment Facility (MTF)	Char(4)	MTF	encounter.location_cd	Join to MDR Location table where the location_cd matches and return mtf. If ERSA = 1, then mtf = substr(place_of_svc_org,1,4).
Patient OHI Flag	Char(1)	OHI	encouter.financial_class	Join to code value dataset using code_set = 354 and active_ind = 1
Person File Primary Key	Num(8)	PERSON_ID	encounter.person_id	No transformation.
Person File Primary Key	Char(100)	PERSON_SK	encounter.person_id	No transformation.
Pre Admit Date & Time	Num(8)	PRE_ADMIT_DT_TM	encounter.pre_reg_dt_tm	Converted to local time.
Patient Identification Process (PIP) Profile	Char(40)	PROFILE	encounter.person_plan_profile_type_cd	Join to code_value table using codeset 368 and active_ind =1.
Provider Role For Admitting Physician (MHS GENESIS)	Char(40)	PROV_ROLE_ADM	encntr_prsnl_reltn. encntr_prsnl_r_cd	Join to the encntr_prsnl_reltn table on encntr_id to retrieve the encntr_prsnl_r_cd. Then join to code_value table where encntr_prsnl_r_cd matches the code_value and code_set = 333 and active_ind = 1 and retrieve display as prov_role.
Provider Role For Attending Physician (MHS GENESIS)	Char(40)	PROV_ROLE_ATT	encntr_prsnl_reltn. encntr_prsnl_r_cd	Join to the encntr_prsnl_reltn table on encntr_id to retrieve the encntr_prsnl_r_cd. Then join to code_value table where encntr_prsnl_r_cd matches the code_value and code_set = 333 and active_ind = 1 and retrieve display as prov_role.
Provider Role For Referring Physician (MHS GENESIS)	Char(40)	PROV_ROLE_REF	encntr_prsnl_reltn. encntr_prsnl_r_cd	Join to the encntr_prsnl_reltn table on encntr_id to retrieve the encntr_prsnl_r_cd. Then join to code_value table where encntr_prsnl_r_cd matches the code_value and code_set = 333 and active_ind = 1 and retrieve display as prov_role.

Field	Format	SAS Name	Source Element	Transformation Rule
Provider Role For Additional Providers (MHS GENESIS)	Char(40)	PROV_ROLE1-PROV_ROLE10	encntr_prsnl_reltn. encntr_prsnl_r_cd	Join to the encntr_prsnl_reltn table on encntr_id to retrieve the encntr_prsnl_r_cd. Then join to code_value table where encntr_prsnl_r_cd matches the code_value and code_set = 333 and active_ind = 1 and retrieve display as prov_role. If the prov_role = 'Attending Provider', set prov_role1 = prov_role. Fill remaining open prov_role array values (up to 10) based on sorted skill_type.
Provider HIPAA Taxonomy-Based Skill Type for Admitting Provider (DMHRSi)	Char(1)	PROV_SKILLD_ADM	prov_hipaa_att	Apply SKILLTYPE&fy.H format from /mdr/ref/caper.hskilltype.fy&fy..txt to PROV_HIPAA_ADM
Provider HIPAA Taxonomy-Based Skill Type for Attending Provider (DMHRSi)	Char(1)	PROV_SKILLD_ATT	prov_hipaa_att	Apply SKILLTYPE&fy.H format from /mdr/ref/caper.hskilltype.fy&fy..txt to PROV_HIPAA_ATT
Provider HIPAA Taxonomy-Based Skill Type for Referring Provider (DMHRSi)	Char(1)	PROV_SKILLD_REF	prov_hipaa_att	Apply SKILLTYPE&fy.H format from /mdr/ref/caper.hskilltype.fy&fy..txt to PROV_HIPAA_REF
Provider HIPAA Taxonomy-Based Skill Type for Admitting Provider (DMHRSi)	Char(1)	PROV_SKILLD1-PROV_SKILLD10	prov_hipaa_att	Apply SKILLTYPE&fy.H format from /mdr/ref/caper.hskilltype.fy&fy..txt
Reason for Visit	Char(255)	REASON_FOR_VISIT	Encounter.REASON_FOR_VISIT	No tranformation
Update Date and Time	Num(8)	UPDT_DT_TM	encounter.updt_dt_tm	No transformation.
Encounter Info				
BENCAT of Record (MHS GENESIS)	Char(40)	BENCAT_E	ei_ben.value_cd	Join genesis_vw.code_value on Encounter Info.value_cd, code_set = 100070 and active_ind = 1

Field	Format	SAS Name	Source Element	Transformation Rule
Patient Category (MHS GENESIS ENCOUNTER)	Char(40)	PATCAT_E	encntr_info.value_cd	Join to the encntr_info table on encntr_id where the info_sub_type_cd = 109901051 and retrieve the value_cd. Then join to code_value table where value_cd matches the code_value and code_set = 100075 and active_ind = 1 and retrieve display.
Encounter Personnel Relation				
Provider ID For Admitting Physician (MHS GENESIS)	Num(8)	PRSNL_ID_ADM	encntr_prsnl_reltn.prsnl_person_id	No tranformation
Provider ID For Attending Physician (MHS GENESIS)	Num(8)	PRSNL_ID_ATT	encntr_prsnl_reltn.prsnl_person_id	No tranformation
Provider ID For Referring Physician (MHS GENESIS)	Num(8)	PRSNL_ID_REF	encntr_prsnl_reltn.prsnl_person_id	No tranformation
Encounter Plan Relation				
Provider ID For Additional Providers (MHS GENESIS)	Num(8)	PRSNL_ID1-PRSNL_ID10	encntr_prsnl_reltn.prsnl_person_id	No tranformation
DHMRSI				
Sponsor Person ID	Num(8)	SPONSOR_PERSON_ID	encntr_plan_reltn.person_id	Join to enctr_plan_reltn table on encntr_id and return person_id.
Provider HIPAA Taxonomy-Based Skill Type Suffix for Admitting Provider (DMHRSi)	Char(1)	PROV_SKSFXD_ADM	dhmrsi.provsksfxnm	Join to DHMRSi by prov_edipn and prov_npi to return provsksfxnm
Provider HIPAA Taxonomy-Based Skill Type Suffix for Attending Provider (DMHRSi)	Char(1)	PROV_SKSFXD_ATT	dhmrsi.provsksfxnm	Join to DHMRSi by prov_edipn and prov_npi to return provsksfxnm

Field	Format	SAS Name	Source Element	Transformation Rule
Provider HIPAA Taxonomy-Based Skill Type Suffix for Referring Provider (DMHRSi)	Char(1)	PROV_SKSFXD_REF	dhmrsi.provsksfxnm	Join to DHMRSi by prov_edipn and prov_npi to return provsksfxnm
Diagnosis				
Admitting Diagnosis Code	Char(7)	ADMDX	diagnosis.adm_dx1	No Transformation
DMHRSI				
Provider HIPAA Taxonomy-Based Skill Type Suffix for Additional Providers	Char(1)	PROV_SKSFXD1-PROV_SKSFXD10	dhmrsi.provsksfxnm	Join to DHMRSi by prov_edipn and prov_npi to return provsksfxnm
Provider Category For Admitting Physician (DMHRSi)	Char(22)	PROV_CATD_ADM	dmhrsi.provcatnm	Join to the MDR DMHRSi table on prov_edipn_adm or prov_npi_adm and return the Personnel Category if the visit date on the encounter record is between the begin and end date associated with the Personnel Category of the provider.
Provider Category For Attending Physician (DMHRSi)	Char(22)	PROV_CATD_ATT	dmhrsi.provcatnm	Join to the MDR DMHRSi table on prov_edipn_att or prov_npi_att and return the Personnel Category if the visit date on the encounter record is between the begin and end date associated with the Personnel Category of the provider.
Provider Category For Referring Physician (DMHRSi)	Char(22)	PROV_CATD_REF	dmhrsi.provcatnm	Join to the MDR DMHRSi table on prov_edipn_ref or prov_npi_ref and return the Personnel Category if the visit date on the encounter record is between the begin and end date associated with the Personnel Category of the provider.
Provider Category for Additional Providers	Char(22)	PROV_CATD1-PROV_CATD10	dmhrsi.provcatnm	Join to the MDR DMHRSi table on prov_edipn[#] or prov_npi[#] and return the Personnel Category if the visit date on the encounter record is between the begin and end date associated with the Personnel Category of the provider.

Field	Format	SAS Name	Source Element	Transformation Rule
Provider Assigned MEPRS Code For Admitting Physician (DMHRSi)	Char(4)	PROV_MEPRSD_ADM	dmhrsi.provmeprnm	Join to the MDR DMHRSi table on prov_edipn_adm or prov_npi_adm and return the Admitting MEPRS if the visit date on the encounter record is between the begin and end date associated with the Assigned MEPRS of the provider.
Provider Assigned MEPRS Code For Attending Physician (DMHRSi)	Char(4)	PROV_MEPRSD_ATT	dmhrsi.provmeprnm	Join to the MDR DMHRSi table on prov_edipn_att or prov_npi_att and return the Assigned MEPRS if the visit date on the encounter record is between the begin and end date associated with the Assigned MEPRS of the provider.
Provider Assigned MEPRS Code For Referring Physician (DMHRSi)	Char(4)	PROV_MEPRSD_REF	dmhrsi.provmeprnm	Join to the MDR DMHRSi table on prov_edipn_ref or prov_npi_ref and return the Assigned MEPRS if the visit date on the encounter record is between the begin and end date associated with the Assigned MEPRS of the provider.
Provider Assigned MEPRS Code For Additional Providers (DMHRSi)	Char(4)	PROV_MEPRSD1 - PROV_MEPRSD10	dmhrsi.provmeprnm	Join to the MDR DMHRSi table on prov_edipn[#] or prov_npi[#] and return the Assigned MEPRS Code if the visit date on the encounter record is between the begin and end date associated with the Assigned MEPRS Code of the provider.
Provider Assigned MTF For Admitting Physician (DMHRSi)	Char(4)	PROV_MTFD_ADM	dmhrsi.provmtfnm	Join to the MDR DMHRSi table on prov_edipn_adm, or prov_npi_admand return the Assigned MTF if the visit date on the encounter record is between the begin and end date associated with the Assigned MTF of the provider.
Provider Assigned MTF For Attending Physician (DMHRSi)	Char(4)	PROV_MTFD_ATT	dmhrsi.provmtfnm	Join to the MDR DMHRSi table on prov_edipn_att or prov_npi_att and return the Assigned MTF if the visit date on the encounter record is between the begin and end date associated with the Assigned MTF of the provider.
Provider Assigned MTF For Referring Physician (DMHRSi)	Char(4)	PROV_MTFD_REF	dmhrsi.provmtfnm	Join to the MDR DMHRSi table on prov_edipn_ref or prov_npi_ref and return the Assigned MTF if the visit date on the encounter record is between the begin and end date associated with the Assigned MTF of the provider.

Field	Format	SAS Name	Source Element	Transformation Rule
Provider Assigned MTF For Additional Providers	Char(4)	PROV_MTFD1 - PROV_MTFD10	dmhrsi.provmtfnm	Join to the MDR DMHRSi table on prov_edipn[#] or prov_npi[#] and return the Assigned MTF if the visit date on the encounter record is between the begin and end date associated with the Assigned MTF of the provider.
Provider Assigned Organization For Admitting Physician (DMHRSi)	Char(8)	PROV_ORGD_ADM	dmhrsi.orgnm	Join to the MDR DMHRSi table on prov_edipn_adm or prov_npi_adm and return the Assigned Organization if the visit date on the encounter record is between the begin and end date associated with the Assigned Organization of the provider.
Provider Assigned Organization For Attending Physician (DMHRSi)	Char(8)	PROV_ORGD_ATT	dmhrsi.orgnm	Join to the MDR DMHRSi table on prov_edipn_att or prov_npi_att and return the Assigned Organization if the visit date on the encounter record is between the begin and end date associated with the Assigned Organization of the provider.
Provider Assigned Organization For Referring Physician (DMHRSi)	Char(8)	PROV_ORGD_REF	dmhrsi.orgnm	Join to the MDR DMHRSi table on prov_edipn_ref or prov_npi_ref and return the Assigned Organization if the visit date on the encounter record is between the begin and end date associated with the Assigned Organization of the provider.
Provider Assigned Organization For Additional Providers	Char(8)	PROV_ORGD1-PROV_ORGD10	dmhrsi.orgnm	Join to the MDR DMHRSi table on prov_edipn[#] or prov_npi[#] and return the Assigned Organization if the visit date on the encounter record is between the begin and end date associated with the Assigned Organization of the provider.
Provider Assigned Service For Admitting Physician (DMHRSi)	Char(1)	PROV_SVC_ASSGD_ADM	dmhrsi.provsvcassgnm	Join to the MDR DMHRSi table on prov_edipn_appt or prov_npi_appt and return the Assigned Service if the visit date on the encounter record is between the begin and end date associated with the Assigned Service of the provider.
Provider Assigned Service For Attending Physician (DMHRSi)	Char(1)	PROV_SVC_ASSGD_ATT	dmhrsi.provsvcassgnm	Join to the MDR DMHRSi table on prov_edipn_att or prov_npi_att and return the Assigned Service if the visit date on the encounter record is between the begin and end date associated with the Assigned Service of the provider.

Field	Format	SAS Name	Source Element	Transformation Rule
Provider Assigned Service For Referring Physician (DMHRSi)	Char(1)	PROV_SVC_ASSGD_REF	dmhrsi.provsvcassgnm	Join to the MDR DMHRSi table on prov_edipn_ref or prov_npi_ref and return the Assigned Service if the visit date on the encounter record is between the begin and end date associated with the Assigned Service of the provider.
Provider Assigned Service For Additional Providers (DMHRSi)	Char(1)	PROV_SVC_ASSGD1-PROV_SVC_ASSGD10	dmhrsi.provsvcassgnm	Join to the MDR DMHRSi table on prov_edipn[#] or prov_npi[#] and return the Assigned Service if the visit date on the encounter record is between the begin and end date associated with the Assigned Service of the provider.
Provider Service For Admitting Physician (DMHRSi)	Char(1)	PROV_SVCD_ADM	dmhrsi.svcnm	Join to the MDR DMHRSi table on prov_edipn_adm or prov_npi_adm and return the Provider Service if the visit date on the encounter record is between the begin and end date associated with the Provider Service of the provider.
Provider Service For Attending Physician (DMHRSi)	Char(1)	PROV_SVCD_ATT	dmhrsi.svcnm	Join to the MDR DMHRSi table on prov_edipn_att or prov_npi_att and return the Provider Service if the visit date on the encounter record is between the begin and end date associated with the Provider Service of the provider.
Provider Service For Referring Physician (DMHRSi)	Char(1)	PROV_SVCD_REF	dmhrsi.svcnm	Join to the MDR DMHRSi table on prov_edipn_ref or prov_npi_ref and return the Provider Service if the visit date on the encounter record is between the begin and end date associated with the Provider Service of the provider.
Provider Service For Additional Providers (DMHRSi)	Char(1)	PROV_SVCD1-PROV_SVCD10	dmhrsi.svcnm	Join to the MDR DMHRSi table on prov_edipn[#] or prov_npi[#] and return the Provider Service if the visit date on the encounter record is between the begin and end date associated with the Provider Service of the provider.
Provider Assigned Unit Identification Code (UIC) for Admitting Physician (DMHRSi)	Char(8)	PROV_UICD_ADM	dmhsri.provuicnm	Join to the MDR DMHRSi table on prov_edipn_adm or prov_npi_adm and return the Assigned UIC if the visit date on the encounter record is between the begin and end date associated with the Assigned UIC of the provider.

Field	Format	SAS Name	Source Element	Transformation Rule
Provider Assigned Unit Identification Code (UIC) for Attending Physician (DMHRSi)	Char(8)	PROV_UICD_ATT	dmhsri.provuicnm	Join to the MDR DMHRSi table on prov_edipn_att or prov_npi_att and return the Assigned UIC if the visit date on the encounter record is between the begin and end date associated with the Assigned UIC of the provider.
Provider Assigned Unit Identification Code (UIC) for Referring Physician (DMHRSi)	Char(8)	PROV_UICD_REF	dmhsri.provuicnm	Join to the MDR DMHRSi table on prov_edipn_ref or prov_npi_ref and return the Assigned UIC if the visit date on the encounter record is between the begin and end date associated with the Assigned UIC of the provider.
DMISID Index				
Provider Assigned Unit Identification Code (UIC) for Additional Providers (DMHRSi)	Char(8)	PROV_UICD1-PROV_UICD10	dmhsri.provuicnm	Join to the MDR DMHRSi table on prov_edipn[#] or prov_npi[#] and return the Assigned UIC if the visit date on the encounter record is between the begin and end date associated with the Assigned UIC of the provider.
Enrollment DMIS ID Market	Char(10)	ENR_MARKET	dmisid_index.market	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the market.
Enrollment DMIS ID Network	Char(40)	ENR_NETWORK	dmisid_index.rep_mkt_name	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the rep_mkt_name.
Enrollment Parent MTF	Char(4)	ENR_PARENT	dmisid_index.ubu_par	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the ubu_par.
PPS Enrollment Site Parent	Char(4)	ENR_PPS_PARENT	dmisod_index.pps_par	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the pps_par.
Enrollment DMIS ID Readiness Service	Char(1)	ENR_READINESS_SVC	dmisod_index.readiness_svc	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the readiness_svc.
Enrollment DMIS ID Branch of Service	Char(1)	ENR_SVC	dmisid_index.ubu_svc	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the ubu_svc.
Enrollment DMIS ID T17 Region	Char(2)	ENR_T17_REG	dmisid_index.t17_reg	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the t17_reg.

Field	Format	SAS Name	Source Element	Transformation Rule
Enrollment DMIS ID T3 Region	Char(2)	ENR_T3_REG	dmisid_index.t3_reg	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the t3_reg.
Treatment MTF MSMA	Char(8)	MTF_CMND	dmisid_index.majcmnd	Join to the DMISID Index table where the treatment mtf matches the dmisid and retrieve the majcmnd.
Treatment MTF Market	Char(10)	MTF_MARKET	dmisid_index.rep_mkt_code	Join to the DMISID Index table where the treatment mtf matches the dmisid and retrieve the rep_mkt_code.
Treatment Parent MTF	Char(3)	MTF_MSMA	dmisid_index.msm_id	Join to the DMISID Index table where the treatment mtf matches the dmisid and retrieve the msm_id.
Treatment MTF Network	Char(40)	MTF_NETWORK	dmisid_index.rep_mkt_name	Join to the DMISID Index table where the treatment mtf matches the dmisid and retrieve the rep_mkt_name.
Treatment Parent MTF (MHS GENESIS)	Char(4)	MTF_PARENT	dmisid_index.ubu_par	Join to the DMISID Index table where the treatment mtf matches the dmisid and retrieve the ubu_par.
Treatment MTF Readiness Service	Char(1)	MTF_READINESS_SVC	dmisid_index.READINESS_SVC	Join to the DMISID Index table where the treatment mtf matches the dmisid and retrieve the READINESS_SVC.
Treatment MTF Branch of Service	Char(1)	MTF_SVC	dmisid_index.ubu_svc	Join to the DMISID Index table where the treatment mtf matches the dmisid and retrieve the ubu_svc.
Treatment MTF T17 Region	Char(2)	MTF_T17_REG	dmisid_index.t17_reg	Join to the DMISID Index table where the treatment mtf matches the dmisid and retrieve the t17_reg.
Treatment MTF T3 Region	Char(2)	MTF_T3_REG	dmisid_index.t3_reg	Join to the DMISID Index table where the treatment mtf matches the dmisid and retrieve the t3_reg.
Primary Care Manager (PCM) Location CD	Char(4)	PCMLOC	dmisid_index.pcmloc	No transformation
PPS Treatment MTF Parent	Char(4)	PPS_PARENT	dmisid_index.pps_par	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the pps_par.

Field	Format	SAS Name	Source Element	Transformation Rule
Enrollment DMIS ID T5 Region	Char(2)	ENR_T5_REG	dmisid_index.t5_reg	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the t5_reg. Populate for FY23+.
Treatment MTF T5 Region	Char(2)	MTF_T5_REG	dmisid_index.t5_reg	Join to the DMISID Index table where the treatment mtf matches the dmisid and retrieve the t5_reg. Populate for FY23+.
DRG				
PPS Rate Basis	Char(1)	PPS_RATE_BASIS	dmisid_index.pcaprate	Join to the DMISID Index table where the enrollment enr_mtf matches the dmisid and retrieve the pcaprate.
Health Plan				
MHS GENESIS MDC	Char(3)	GEN_MDC	dx1	Join the drg.mdc_cd to code_value where code_set = 14285 and active_ind = 1
LVM				
Patient Identification Process (PIP) Plan Name	Char(150)	PIP_PLAN_NAME	health_plan.plan_name	Join to enctr_plan_reltn table on encntr_id, then join to health plan table on health_plan_id where active_ind = 1 and return plan_name.
Alternate Care Value (ACV)	Char(1)	ACV	lvm.acv	Fill with ACV if the visit date on the encounter record is between the begin and end date associated with the ACV.
Alternate Care Value (ACV) Group	Char(2)	ACVGROUP	lvm.acvgrp	Blank for all records after Jan 1, 2019. Derived from ACV and comben (before 1/1/18) or Enrollment Group, PCM type, Eligibility group, and comben (after 1/1/18).
DEERS Beneficiary Category	Char(3)	BENCAT	lvm.r_ben_cat_cd	Fill with bencat associated with this EDIPN. If there is no match for this patient in the LVM, set to UNK.
DEERS Common Beneficiary Category	Char(1)	COMBEN	lvm.cben	Derived from Beneficiary Category during LVM merge: 1 = Dep Active Duty / Guard 2 = Retired 3 = Dep of Retired / Survivor / Other / Unknown / IGR / IDG 4 = Active Duty / Guard

Field	Format	SAS Name	Source Element	Transformation Rule
DEERS Eligibility Group	Char(1)	ELG_GRP	lvm.elggrp	Fill with Eligibility Group if the visit date on the encounter record is between the begin and end date associated with the Eligibility Group.
DEERS Enrollment Group	Char(1)	ENR_GRP	lvm.enrgrp	Fill with Enrollment Group if the visit date on the encounter record is between the begin and end date associated with the Enrollment Group.
Enrollment DMIS ID	Char(4)	ENR_MTF	lvm.enr	Fill with Enrollment MTF if the visit date on the encounter record is between the begin and end date associated with the enrollment of the patient to the MTF.
Ethnic Background Code from DEERS	Char(1)	ETHNIC_DEERS	lvm.ethnic	Fill with Ethnic DEERs if the visit date on the encounter record is between the begin and end date associated with the Ethnic DEERs.
Gender	Char(1)	GENDER	lvm.gender	Fill with Gender if the visit date on the encounter record is between the begin and end date associated with the Gender.
Assigned HCDP Code	Char(3)	HCDP_ASSGN	lvm.asghcdp	Fill with assigned HCDP code if the visit date on the encounter record is between the begin and end date associated with the assigned HCDP code.
HCDP Code	Char(3)	HCDPLVM	lvm.hcdp	Fill with enrollment HCDP code if the visit date of the encounter record is between the begin and end date associated with the enrollment HCDP code.
DEERS Marital Status	Char(1)	MARITAL	lvm.ms	Fill with Marital Status if the visit date on the encounter record is between the begin and end date associated with the Marital Status.
Medicare Flag	Char(1)	MEDICARE_FLAG	lvm.mf	Fill with Medicare Eligibility Code if the visit date on the encounter record is between the begin and end date associated with the Medicare Eligibility Code.
DEERS Patient Zip Code	Char(5)	PATZIP	lvm.zip	Fill with ZIP Code if the visit date on the encounter record is between the begin and end date associated with the ZIP Code.

Field	Format	SAS Name	Source Element	Transformation Rule
Primary Care Manager (PCM) Type	Char(1)	PCM_TYPE	lvm.pcmtyp	Fill with Enrollment PCM Type if the visit date on the encounter record is between the begin and end date associated with the Enrollment PCM Type.
Primary Care Manager (PCM) Provider ID	Char(18)	PCMIDLVM	lvm.pcm	Fill with PCM ID if the visit date on the encounter record is between the begin and end dates associated with the PCM ID.
DEERS Sponsor Service, Aggregate	Char(1)	SAGGLVM	lvm.aggsvc	Fill with Sponsor Service Aggregated if the visit date on the encounter record is between the begin and end date associated with the Sponsor Service Aggregated. If the visit date is outside of the dates associated with the Sponsor Service, or there is no match for this patient in the LVM, set to Z.
DEERS Sponsor Service	Char(1)	SSVCLVM	lvm.svc	Fill with Sponsor Service if the visit date on the encounter record is between the begin and end date associated with the Sponsor Service. If the visit date is outside of the dates associated with the Sponsor Service, or there is no match for this patient in the LVM, set to Z.
TPR Eligibility Flag	Char(1)	TPRELIG	lvm.tpr	Fill with TRICARE Prime Remote status if the visit date on the encounter record is between the begin and end date associated with the TPR status.
TRICARE Young Adult Flag	Char(1)	TYAFLAG	lvm.tya	Fill with TRICARE Young Adult status if the visit date on the encounter record is between the begin and end date associated with the TYA status.
MDR Appointment				
Patient Privilege Code	Char(1)	PRIVILEGE	lvm.priv	Fill with Medical Privilege Code if the visit date on the encounter record is between the begin and end date associated with the Medical Privilege Code.

Field	Format	SAS Name	Source Element	Transformation Rule
Appointment Made Date	Num(8)	APPT_MADE_DT_TM	sch_event_action.action_dt_tm	Join to sch_event_action on sch_event_id where sch_action_cd = 4517 and active_ind = 1 and action_meaning = 'SCHEDULE' and retrieve the latest action_dt_tm value. If the value is missing, set to missing. If populated and earlier than the appt_dt_tm, set to action_dt_tm. Otherwise, set to the appt_dt_tm. Converted to local time.
Provider ID For Appointment Physician (MHS Genesis)	Char(100)	PROVID_APPT	mdr_appointment.provid1	No transformation.
MDR Diagnosis				
Scheduled Appt Key	Char(100)	SCH_APPOINTMENT_SK	MDR_Appointment.sch_appointment_sk	No transformation
Diagnosis Contributor System (MHS GENESIS)	Char(60)	CONTRIBUTOR_SYSTEM_DX	mdr_diagnosis.CONTRIBUTOR_SYSTEM_DX	
CCE Diagnosis Code 1 - 50	Char(7)	DX_CCE1-DX_CCE50	DX_CCE1	No Transformation
PowerChart Diagnosis Code 1-50	Char(7)	DX_PC1-DX_PC50	DX_PC1	No Transformation
Diagnosis Code 1 - 50 (Combination of CCE and PC diagnosis codes)	Char(7)	DX1-DX50	dx1	No Transformation
Present on Admission Indicator 1-50	Char(1)	DX1POA-DX50POA	DX1POE	No Transformation
MDR Location				
Admission Location (General)	Char(60)	ADMISSION_LOC_GENERAL	Mdr_location.display	No Transformation
Clinic State of Record	Char(2)	CLINSTAT_R	mdr_location.state	Join to the MDR GENESIS Location table where the current_loc matches the location_sk and return the state.
Clinic Zip Code of Record	Char(25)	CLINZIP_R	mdr_location.postal_code	Join to the MDR GENESIS Location table where the current_loc matches the location_sk and return the postal_code.

Field	Format	SAS Name	Source Element	Transformation Rule
GENESIS Data Begin Date	Num(8)	GEN_BEGIN_DT	mdr_location.gen_begin_dt	Join to the MDR GENESIS Location table where the current_loc matches the location_sk and return the GENESIS Begin Date. Format date as MMDDYY10.
Treatment Parent MTF (DMIS ID Index)	Char(4)	MTF_PARENT_REC	mdr_location.mtf_parent	Join to the MDR GENESIS Location table where the current_loc matches the location_sk and return the mtf_parent.
MEPRS Reporting Status of MTF	Num(8)	NOMEPRS_FLAG	mdr_location.nomeprs_flag	Join to the MDR GENESIS Location table where the current_loc matches the location_sk and return the nomeprs_flag.
MDR Person				
Test Location Flag	Num(8)	TEST_LOCATION_FLAG	MDR_Location.test_record_ind	No transformation
Ethnic Background Code	Char(1)	ETHNIC	mdr_person.ethnic	Join to the MDR GENESIS Person table on person_sk and retrieve the ethnic value.
Beneficiary First Name	Char(20)	FIRSTNAME	mdr_person.FIRST_NAME	No transformation.
Gender, Raw	Char(10)	GENDER_R	mdr_person.gender	Join to the MDR GENESIS Person table on person_sk and retrieve the gender.
Beneficiary Last Name	Char(26)	LASTNAME	mdr_person.last_name	Join to the MDR GENESIS Person table on person_sk and retrieve the last_name.
Medical Record Number (Patient)	Char(40)	MRN	mdr_person.mrn	Join to the MDR GENESIS Person table on person_sk and retrieve the mrn.
Person Association Reason Code	Char(2)	PARC	mdr_person.parc	Join to the MDR GENESIS Person table on person_sk and retrieve the parc.
DEERS Patient Date of Birth	Char(8)	PATDOB	mdr_person.birth_dt	Join to the MDR GENESIS Person table on person_sk and retrieve the birth_dt.
Beneficiary Patient Name	Char(74)	PATNAME	person.name_full_formatted	Join to the person table on person_id and retrieve the name_full_formatted.
Patient Gender	Char(1)	PATSEX	mdr_person.gender	Join to the MDR GENESIS Person table on person_sk and retrieve the gender.
DEERS Patient Social Security Number	Char(9)	PATSSN	mdr_person.ssn	Join to the MDR GENESIS Person table on person_sk and retrieve the ssn.

Field	Format	SAS Name	Source Element	Transformation Rule
Sponsor Social Security Number	Char(9)	SPONSSN	mdr_person.sponssn	Join to the MDR GENESIS Person table on person_sk and retrieve the sponssn.
MDR Personnel				
Test Person Flag	Num(8)	TEST_PERSON_FLAG	MDR_Person.test_record_ind	No transformation
Provider EDIPN for Additional Providers	Char(10)	PROV_EDIPN1-PROV_EDIPN10	mdr_personnel.prsnl_edipn	Join to the MDR GENESIS Personnel table where the provid[#] matches the prsnl_id and retrieve the prsnl_edipn.
Primary HIPAA Taxonomy For Additional Providers	Char(10)	PROV_HIPAA1-PROV_HIPAA10	mdr_personnel.hipaa1	Join to the MDR GENESIS Personnel table where the provid[#] matches the prsnl_id and retrieve the hipaa1 value.
Provider Name, Admitting	Char(100)	PROV_NAME_ADM	provid_att, name_full_formatted	Join to the prsnl table where provid_att = person_id. Then return the name_full_formatted field.
Provider Name, Appointment	Char(100)	PROV_NAME_APPT	prsnl.name_full_formatted	Join to the prsnl table on provid_appt = prsnl.person_id and retrieve the name_full formatted.
Provider Name, Attending	Char(100)	PROV_NAME_ATT	provid_att, name_full_formatted	Join to the prsnl table where provid_att = person_id. Then return the name_full_formatted field.
Provider Name, Referring	Char(100)	PROV_NAME_REF	provid_ref, name_full_formatted	Join to the prsnl table where provid_ref = person_id. Then return the name_full_formatted field.
Provider Name, Additional Providers	Char(100)	PROV_NAME1 - PROV_NAME10	mdr_personnel.full_name	Join to the MDR GENESIS Personnel table where the provid[#] matches the prsnl_id and retrieve the full_name.
Provider NPI For Admitting Physician	Char(10)	PROV_NPI_ADM	admitting_prov_id, alias	Join to the prsnl table where admitting_prov_id = person_id. Then join to the prsnl_alias table on admitting_prov_id where the alias_pool_cd = 4038127 and active_ind = 1 and return the alias value.
Provider NPI For Appointment Physician	Char(10)	PROV_NPI_APPT	prsnl_alias.alias	Join to the prsnl_alias table on prsnl.person_id = prsnl_alias.person_id where the prsnl_alias_type_cd = 4038127 and active_ind = 1 and retrieve the alias value

Field	Format	SAS Name	Source Element	Transformation Rule
Provider NPI For Attending Physician	Char(10)	PROV_NPI_ATT	provid_att, alias	Join to the prsnl table where provid_att = person_id. Then join to the prsnl_alias table on provid_att where the alias_pool_cd = 4038127 and active_ind = 1 and return the alias value.
Provider NPI For Referring Physician	Char(10)	PROV_NPI_REF	provid_att, alias	Join to the prsnl table where provid_ref = person_id. Then join to the prsnl_alias table on provid_ref where the alias_pool_cd = 4038127 and active_ind = 1 and return the alias value.
Provider NPI For Additional Providers	Char(10)	PROV_NPI1 - PROV_NPI10	mdr_personnel.npi	Join to the MDR GENESIS Personnel table where the provid[#] matches the prsnl_id and retrieve the NPI.
Provider NPI Type For Admitting Physician	Char(1)	PROV_NPITYPE_ADM	provid_att, npitype	Join to the prsnl table where provid_att = person_id. Then return the npitype field.
Provider NPI Type For Appointment Physician	Char(1)	PROV_NPITYPE_APPT	prsnl.npitype	Join to the prsnl table on provid_appt = prsnl.person_id and retrieve the npitype.
Provider NPI Type For Attending Physician	Char(1)	PROV_NPITYPE_ATT	provid_att, npitype	Join to the prsnl table where provid_att = person_id. Then return the npitype field.
Provider NPI Type For Referring Physician	Char(1)	PROV_NPITYPE_REF	provid_ref, npitype	Join to the prsnl table where provid_ref = person_id. Then return the npitype field.
Provider NPI Type For Additional Providers	Char(1)	PROV_NPITYPE1-PROV_NPITYPE10	mdr_personnel.npitype	Join to the MDR GENESIS Personnel table where the provid[#] matches the prsnl_id and retrieve the NPITYPE.
Skill Type based on Primary HIPAA Taxonomy For Admitting Physician	Char(1)	SKILLH_ADM	mdr_personnel.skill_type	Join to the MDR GENESIS Personnel table where the admitting provid matches the prsnl_id and retrieve the skill_type field.
Skill Type based on Primary HIPAA Taxonomy For Appointment Physician	Char(1)	SKILLH_APPT	mdr_personnel.skill_type	Join to the MDR GENESIS Personnel table where the appointing provid matches the prsnl_id and retrieve the skill_type field.

Field	Format	SAS Name	Source Element	Transformation Rule
Skill Type based on Primary HIPAA Taxonomy For Attending Physician	Char(1)	SKILLH_ATT	mdr_personnel.skill_type	Join to the MDR GENESIS Personnel table where the attending provid matches the prsnl_id and retrieve the skill_type field.
Skill Type based on Primary HIPAA Taxonomy For Referring Physician	Char(1)	SKILLH_REF	mdr_personnel.skill_type	Join to the MDR GENESIS Personnel table where the referring provid matches the prsnl_id and retrieve the skill_type field.
Skill Type based on Primary HIPAA Taxonomy For Additional Providers	Char(1)	SKILLH1-SKILLH10	mdr_personnel.skill_type	Join to the MDR GENESIS Personnel table where the provid[#] matches the prsnl_id and retrieve the skill_type field.
Test Personnel Flag - Admitting Physician	Char(1)	TEST_RECORD_ADM	MDR_Personnel.test_record_ind	No transformation
Test Personnel Flag - Appointment Physician	Char(1)	TEST_RECORD_APPT	MDR_Personnel.test_record_ind	No transformation
Test Personnel Flag - Attending Physician	Char(1)	TEST_RECORD_ATT	MDR_Personnel.test_record_ind	No transformation
Test Personnel Flag - Referring Physician	Char(1)	TEST_RECORD_REF	MDR_Personnel.test_record_ind	No transformation
Test Personnel Flag - Additional Provider 1-10	Char(1)	TEST_RECORD1-TEST_RECORD10	MDR_Personnel.test_record_ind	No transformation
Organization				
Vocabulary DRG	Char(60)	VOCABULARY_DRG	source_vocabulary_cd	code_set = 400 and active_ind = 1
Patient Identification Process (PIP) Sponsor Branch	Char(150)	PIP_SPONSOR_BRANCH	organization.org_name	Join to enctr_plan_reltn table on encntr_id, then join to person_org_reltn where sponsor_person_org_reltn_id = person_org_reltn_id, then join to organization on organization_id and return org_name.

Field	Format	SAS Name	Source Element	Transformation Rule
Person Info				
Patient Category (MHS GENESIS PERSON)	Char(40)	PATCAT_P	person_info.value_cd	Join to the person_info table on person_id where the info_sub_type_cd = 114540103 and retrieve the value_cd. Then join to code_value table where value_cd matches the code_value and code_set = 100075 and active_ind = 1 and retrieve display.
Person Org Reltn				
Patient Identification Process (PIP) Sponsor Grade	Char(60)	PIP_SPONSOR_GRADE	person_org_reltn.empl_type_cd	Join to enctr_plan_reltn table on encntr_id, then join to person_org_reltn where sponsor_person_org_reltn_id = person_org_reltn_id to obtain empl_type_cd. Join to code_value table where empl_type_cd matches the code_value and code_set = 26 and active_ind = 1 and retrieve display.
Patient Identification Process (PIP) Sponsor Rank	Char(60)	PIP_SPONSOR_RANK	person_org_reltn.empl_title_cd	Join to enctr_plan_reltn table on encntr_id, then join to person_org_reltn where sponsor_person_org_reltn_id = person_org_reltn_id to obtain empl_title_cd. Join to code_value table where empl_title_cd matches the code_value and code_set = 376 and active_ind = 1 and retrieve display.
Procedure				
Patient Identification Process (PIP) Sponsor Status	Char(60)	PIP_SPONSOR_STATUS	person_org_reltn.empl_occupation_cd	Join to enctr_plan_reltn table on encntr_id, then join to person_org_reltn where sponsor_person_org_reltn_id = person_org_reltn_id to obtain empl_occupation_cd. Join to code_value table where empl_occupation_cd matches the code_value and code_set = 374 and active_ind = 1 and retrieve display.
Schedule Appointment				
Procedure Code (1-50)	Num(8)	PROC_1-PROC_50	procedure.proc	Procedure on the record sorted by encntr_id proc_priority.
Appointment Date and Time	Num(8)	APPT_DT_TM	sch_appt.beg_dt_tm	Converted to local time.

Field	Format	SAS Name	Source Element	Transformation Rule
Appointment Status	Char(20)	APPT_STAT	sch_appt.sch_state_cd	Join to sch_appt on encntr_id and retrieve sch_state_cd. Then join to code_value table where sch_state_cd matches the code_value and code_set = 14233 and active_ind = 1 and retrieve display. Convert text to upper case.
Internally Derived Fields				
Assigned Duration	Char(3)	ASSGNDUR	sch_appt.duration	Join to sch_appt on encntr_id where sch_state_cd in (4536,4537) and sch_role_cd = 4572 and active_ind = 1 and retrieve duration.
Admission Inferred Flag (MHS GENESIS)	Num(8)	ADM_INFR_FLAG	discharge_dt_tm, dx_cce1	if discharge_dt_tm = . or dx_cce1 = '' then ADM_INFR_FLAG = 1; else ADM_INFR_FLAG = 0;
Admission Location Type of Care	Char(10)	ADMISSION_LOC_COMPCARE	admission_loc_composite	Pull out the third segment of the nurse unit location code
Admission Location Specialty Type	Char(10)	ADMISSION_LOC_COMPSPEC	admission_loc_composite	Pull out the second segment of the nurse unit location code
Age Group	Char(1)	AGEGRP	patage	If patage is 0-4, set to A; if 5-14, B; if 18-24, C; if 25-34, D; if 35-44, F; if 45-64, G; if 65+, H; else X.
Legacy Appointment Status	Char(2)	APPT_STAT_LEGACY	appt_stat	If appt_stat eq 'CANCELED' then appt_stat_legacy = '3'; else if appt_stat eq 'NO SHOW' then appt_stat_legacy = '4'; else if appt_stat in ('CHECKED OUT' 'CHECKED IN') then do; if walk_in_flag eq '1' then appt_stat_legacy = '5'; else if walk_in_flag eq '0' then appt_stat_legacy = '2'; end; else if appt_stat in ('CONFIRMED' 'SCHEDULED') then appt_stat_legacy = '12'; else appt_stat_legacy = '14';

Field	Format	SAS Name	Source Element	Transformation Rule
Appointment Type	Char(16)	APPT_TYPE	appt_type_txt	From the appointment_type_txt value, use regular expression pattern matching to derive the appt_type.
Legacy Appointment Type	Char(6)	APPT_TYPE_LEGACY	appt_type	Apply the $appttyplegacy. format to map the appt_type values to the appt_type_legacy_values.
Appointment Type Text	Char(255)	APPT_TYPE_TXT	appt_type_txt	If substr(appt_type_txt,1,7) = 'Walk-In' then set the walk_in_flag = 1, otherwise set it to 0.
Appointment Prefix (Source System Flag)	Char(1)	APPTPFIX		Set to M for all records.
Bed Days	Num(8)	BED_DAYS	Days_1-Days_30	Sum of DAYS_1-DAYS_30
Beneficiary T17 Region	Char(2)	BEN_T17_REG	patzip	Based on matching FY, FM and patzip; Set equal to T17_REG. If patzip not found in MDR Omni-CAD, leave blank.
Beneficiary T3 Region	Char(2)	BEN_T3_REG	patzip	Based on matching FY, FM and patzip; Set equal to T3_REG. If patzip not found in MDR Omni-CAD, leave blank.

Calculated Bed Days	Num(8)	CALCDAYS	discharge_dt, adm_dt	MTF_DAYS = discharge_dt - adm_dt;
				if mtf_days = 0 then CALCDAYS = 1;
				else CALCDAYS = mtf_days;

Beneficiary Catchment Area	Char(4)	CATCH	patzip, sagglvm	Based on matching FY, FM and patzip; if sagglvm = A then set equal to AWORLD, if sagglvm = F then set equal to FWORLD; if sagglvm in (M, N, V) then set equal to NWORLD, otherwise set equal to OWORLD.
Calendar Month, Admission	Char(2)	CMADM	inpatient_admit_dt_tm and reg_dt_tm	Extract the calendar month from inpatient_adm_dt_tm (if populated), otherwise from reg_dt_tm value.
Calendar Month, Discharge	Char(2)	CMDISP	discharge_dt_tm	Extract the calendar month from discharge_dt_tm.

Field	Format	SAS Name	Source Element	Transformation Rule
Calendar Year, Admission	Char(4)	CYADM	inpatient_admit_dt_tm and reg_dt_tm	Extract the calendar month from inpatient_adm_dt_tm (if populated), otherwise from reg_dt_tm value.
Calendar Year, Discharge	Char(4)	CYDISP	discharge_dt_tm	Extract the calendar year from discharge_dt_tm.

Days in Segment 1 - 20	Num(8)	DAYS_1-DAYS_20	Encounter_History.start_dt_tm, Encounter_History.end_dt_tm	Derive days and duration in this segment;
				if datepart(end_dt_tm) ge '01JAN2100'd then duration = .;
				else duration = end_dt_tm - start_dt_tm;
				if datepart(end_dt_tm) ge '01JAN2100'd then days = .;
				else if enc_type = 'Inpatient' then days = datepart(end_dt_tm) - datepart(start_dt_tm);
				else days = 0;

Disposition Date	Char(8)	DISPDATE	encounter.disch_dt_tm	Formatted as yymmddn8.
Duration 1- 20 (Encounter History)	Num(8)	DURATION_1-DURATION_20	Encounter_History.start_dt_tm, Encounter_History.end_dt_tm	Days calculation displayed in seconds.
Duration Stay Flag	Num(8)	DURING_STAY_FLAG	during_first_enc_type	If there are any non-Inpatient segments during the stay, set the DURING_STAY_FLAG =1 else 0.
Diagnosis Group	Char(2)	DXGRP	dx1	Use first three characters of Diagnosis 1 (DX1). See Appendix D for derivation rules.
Encounter Type (Encounter History)	Char(26)	ENC_TYPE_1-ENC_TYPE_20	encntr_loc_hist.enc_type	Encounter Type per segment.
Encounter PI-EDW Key	Num(8)	ENCOUNTER_KEY		Placeholder, always blank.
Encounter Key (Primary)	Char(100)	ENCOUNTER_NK	encounter_sk	Concatenate the encounter_sk value and '18365' using the cat function: cat(encounter_sk, '18635')
Ending Transaction Date/Time 1-20 (Enc History)	Num(8)	END_TRANS_DT_1-END_TRANS_DT_20	encntr_loc_hist.transaction_dt_tm	end transaction_dt_tm for segment 1
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 other

Field	Format	SAS Name	Source Element	Transformation Rule
Fiscal Month, Admission	Char(2)	FMADM	inpatient_admit_dt_tm and reg_dt_tm	Extract the fiscal month from inpatient_adm_dt_tm (if populated), otherwise from reg_dt_tm value.
Fiscal Month, Discharge	Char(2)	FMDISP	discharge_dt_tm	Extract the fiscal month from discharge_dt_tm.
Follow Up Flag	Num(8)	FOLLOW_UP_FLAG	start_dt, discharge_dt_tm	Encounter History table, if there are any segments after the discharge date, set the FOLLOW_UP_FLAG
Fiscal Year, Admission	Char(4)	FYADM	inpatient_admit_dt_tm and reg_dt_tm	Extract the fiscal month from inpatient_adm_dt_tm (if populated), otherwise from reg_dt_tm value.
Fiscal Year, Discharge	Char(4)	FYDISP	discharge_dt_tm	Extract the fisacl year from discharge_dt_tm.
MHS GENESIS MSDRG	Char(4)	GEN_MSDRG	MSDRG Grouper	No Transformation.
GENESIS Flag	Num(8)	GENESIS_FLAG		Set to 1 for all records.
Health System ID	Num(8)	HEALTH_SYSTEM_ID		Set to 18635 for all recoreds.
Bed Days in ICU	Num(8)	ICU_DAYS	nurseloc	Where NULC contains ICU then days in ICU.
ICU Flag	Char(1)	ICU_FLAG	nurseloc	Where NULC contains ICU then ICU = Y, else N.
Medical Service 1-20 (Enc History)	Char(43)	MEDSVC_1-MEDSVC_20	encounter_history.medsvc	Medical Service from each segment of the record.
Member Number	Char(100)	MEMBER_NBR	encounter_history.medsvc	Medical Service from each segment of the record.
Treatment MEPRS 1 Code	Char(1)	MEPR1	meprscd	Use a substring function to take the first character of the meprscd: substr(meprscd,1,1)
Treatment MEPRS 2 Code	Char(2)	MEPR2	meprscd	Use a substring function to take the first two characters of the meprscd: substr(meprscd,1,2)
Treatment MEPRS 3 Code	Char(3)	MEPR3	meprscd	Use a substring function to take the three characters of the meprscd: substr(meprscd,1,3)
MEPRS Code 1-20 (Enc History)	Char(3)	MEPR3_1-MEPR3_20	meprscd	Use a substring function to take the three characters of the meprscd: substr(meprscd,1,3) of segment 1
MS-DRG Code	Char(3)	MSDRG	MSDRG Grouper	No Transformation.

Field	Format	SAS Name	Source Element	Transformation Rule
MS-DRG Baseline Relative Weighted Product	Num(8)	MSDRGBASERWP	MSDRG Grouper	No Transformation.
MS-DRG Transfer Status Flag	Num(8)	MSDRGICAT	MSDRG Grouper	No Transformation.
MS-DRG Outlier Relative Weighted Product	Num(8)	MSDRGOUTRWP	MSDRG Grouper	No Transformation.
MS-DRG Return Code	Num(8)	MSDRGRTC	MSDRG Grouper	No Transformation.
MS-DRG Relative Weighted Product	Num(8)	MSDRGRWP	MSDRG Grouper	No Transformation.
Medical/Surgical Indicator	Char(1)	MSDRGSURG	MSDRG Grouper	No Transformation.
MS-DRG Version	Char(4)	MSDRGVRSN	MSDRG Grouper	No Transformation.
MS-DRG Assigned MDC	Char(2)	MSMDC	MSDRG Grouper	No Transformation.
Total Bed Days	Num(8)	MTF_DAYS	discharge_dt, adm_dt	No Transformation.
Patient's MTF Service Area	Char(4)	MTFSVCAREA	patzip, sagglvm	Based on matching FY, FM and patzip; if sagglvm = A then set equal to ABPA, if sagglvm = F then set equal to FBPA; if sagglvm in (M, N, V) then set equal to NBPA, otherwise set equal to OPRISM.
Nurse Location 1-20 (Enc History)	Char(25)	NURSELOC_1-NURSELOC_20	encounter_history.nurseloc	Nurse Unit Location on segment 1
Observation Stay Flag	Char(1)	OBS_FLAG	dx1, mtf_days	Y, when DX1='Z049' and MTF_DAYS<2 N, otherwise
Patient Age	Num(8)	PATAGE	patdob, visit_dt	Derived age in years between the patdob and visit_dt.
PPS Earnings Factor	Num(8)	PPS_EARNINGS_FACTOR		Set to 1 for all records. Formatted as Num(5.3)
Pre Admit Flag	Num(8)	PREADMIT_FLAG	START_TRANS_DT, adm_dt_tm	If there are any segments before the admission date, set the PREADMIT_FLAG (else set it to 0)

Field	Format	SAS Name	Source Element	Transformation Rule
Beneficiary PRISM Area	Char(4)	PRISM	patzip, sagglvm	Based on matching FY, FM and patzip; if sagglvm = A then set equal to APRISM, if sagglvm = F then set equal to FPRISM; if sagglvm in (M, N, V) then set equal to NPRISM, otherwise set equal to OPRISM.
Procedure Code Date & Time (Procedures) 1-50	Num(8)	PROC_DT_TM_1-PROC_DT_TM_50	procedure.proc, procedure.proc_priority	Procedure on the record sorted by proc_priority, Procedure 1
Procedure Code (1-50) and Provider ID (1-5)	Num(8)	PROC_j_PRSNL_ID_k, where j=1-50 and k=1-5	encntr_prsnl_reltn	Procedure 1, Provider ID 1
Procedure Code Priority Sequence 1-50	Num(8)	PROC_PSEQ1-PROC_PSEQ50	procedure.proc_priority	No transformation.
Procedure Code 1-50 Units of Service	Num(8)	PROCQTY_1-PROCQTY_50	procedure.units_of_service	Units of service and multiplyer for the same procedure code showing up more than once for procedure 1
Procedure Code 1-50 Units of Service (RAW)	Num(8)	PROCQTY_RAW_1-PROCQTY_RAW_50	procedure.units_of_service	Units of service for procedure 1
Product Line	Char(2)	PRODLINE	mepr3, mtf_svc	Derived based on Clinic (MEPRS3) and Treatment Service (mtf_svc). See Table A2 for derivation rules.
Provider EDIPN For Admitting Physician	Char(10)	PROV_EDIPN_ADM	admitting_prov_id, alias	Join to the prsnl table where admitting_prov_id = person_id. Then join to the prsnl_alias table on person_id where the alias_pool_cd = 106935631 and prsnl_alias_type_cd = 685806 and alias like '1%' and alias similar to '[0-9]{10}' and active_ind = 1 and retrieve the alias value.

Field	Format	SAS Name	Source Element	Transformation Rule
Provider EDIPN For Appointment Physician	Char(10)	PROV_EDIPN_APPT	prsnl_alias.alias	Join to the prsnl_alias table on prsnl.person_id = prsnl_alias.person_id where the alias_pool_cd = 106935631 and prsnl_alias_type_cd = 685806 and alias like '1%' and alias similar to '[0-9]{10}' and active_ind = 1 and retrieve the alias value.
Provider EDIPN For Attending Physician	Char(10)	PROV_EDIPN_ATT	provid_att, alias	Join to the prsnl table where provid_att = person_id. Then join to the prsnl_alias table on person_id where the alias_pool_cd = 106935631 and prsnl_alias_type_cd = 685806 and alias like '1%' and alias similar to '[0-9]{10}' and active_ind = 1 and retrieve the alias value.
Provider EDIPN For Referring Physician	Char(10)	PROV_EDIPN_REF	provid_ref, alias	Join to the prsnl table where provid_ref = person_id. Then join to the prsnl_alias table on person_id where the alias_pool_cd = 106935631 and prsnl_alias_type_cd = 685806 and alias like '1%' and alias similar to '[0-9]{10}' and active_ind = 1 and retrieve the alias value.
Primary HIPAA Taxonomy For Admitting Physician	Char(10)	PROV_HIPAA_ADM	admitting_prov_id, hipaa1	Join to MDR NPPES table on admitting_prov_npi and return the hipaa1 field.
Primary HIPAA Taxonomy For Appointment Physician	Char(10)	PROV_HIPAA_APPT	nppes.hipaa1	Join to the MDR NPPES table on prov_npi_appt = nppes.npi and retrieve the hipaa1 value.
Primary HIPAA Taxonomy For Attending Physician	Char(10)	PROV_HIPAA_ATT	prov_npi_att, hipaa1	Join to MDR NPPES table on prov_npi_att and return the hipaa1 field.
Primary HIPAA Taxonomy For Referring Physician	Char(10)	PROV_HIPAA_REF	prov_npi_ref, hipaa1	Join to MDR NPPES table on prov_npi_ref and return the hipaa1 field.
Provider ID For Admitting Physician (MHS Genesis)	Char(100)	PROVID_ADM	Encounter_Prsnl_Reltn	Encounter_Prsnl_Reltn = 1126 and Active_ind =1 and prsnl_person_id > 0
Provider ID For Attending Physician (MHS Genesis)	Char(100)	PROVID_ATT	Encounter_Prsnl_Reltn	Encounter_Prsnl_Reltn = 1116 and Active_ind =1 and prsnl_person_id > 0

Field	Format	SAS Name	Source Element	Transformation Rule
Provider ID For Referring Physician (MHS Genesis)	Char(100)	PROVID_REF	Person_Info.info_sub_type_cd	Person_Info.info_sub_type_cd = 114540103 and active_ind = 1
Provider ID For Additional Providers (MHS Genesis)	Char(100)	PROVID1-PROVID10	Encounter_Prsnl_Reltn	Proivder ID for Provider position 1
DEERS Patient Race Code	Char(1)	RACE	bencat, race_genesis, race_deers	If the patient is Active Duty or Guard, use DEERS race value as primary over GENESIS race value, except for 'Other' and 'Unknown'. If the patient is not Active Duty or Guard, use GENESIS race value as primary over DEERS. See Table A3 for full derivation.
Sponsor Branch of Service for M2	Char(1)	RSPONSVC	patcat_e,patcat_p,ssvclvm	Derive from 1st character of PATCAT_E if available. Otherwise, use SSVCLVM. 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. Else assign X.
Service Line	Char(5)	SERVICE_LINE	mepr3	Apply the Service line format from /mdr/ref/slfmt to MEPR3 as follows: put(mepr3,$slfmt.);
Service Line (Segment) 1 - 20	Char(5)	SERVICE_LINE_1-SERVICE_LINE_20	mepr3	Apply the Service line format from /mdr/ref/slfmt to MEPR3 as follows: put(mepr3,$slfmt.); for segment 1
Sponsor Person ID Type Code	Char(1)	SPONSIDTYPE	sponssn	Derive sponsidtype if Sponsor SSN is available; length SPONSIDTYPE $1.; if sponssn ne '' then sponsidtype = 'S'; else sponsidtype = '';
Starting Transaction Date/Time 1-20 (Enc History)	Num(8)	START_TRANS_DT_1-START_TRANS_DT_20	encntr_loc_hist.transaction_dt_tm	Start transaction_dt_tm for segment 1

Field	Format	SAS Name	Source Element	Transformation Rule
Beneficiary T5 Region	Char(2)	BEN_T5_REG	patzip	Based on matching FY, FM and patzip; Set equal to T5_REG. If patzip not found in MDR Omni-CAD, leave blank. Populate for FY23+.

APPENDIX A: Administrative Text Processing Steps and Field Additions

The following processes will be applied to the Admission 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), T5 Region (MTF_T5_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), T5 Region (ENR_T5_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), T5 Region (BEN_T5_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. When the admit date is prior to birth date, set admit date to birth date for grouping only. Do not change the underlying data.
3. 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_CCE50), POA indicators, and procedure codes (PROC_1-PROC_50) are formatted for grouping.
    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.

4. 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_50);

        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_cce50 proc_1-proc_50 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 *;
        renamepatsex = 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[50] $7. proc1-proc50;
        array xproc[50] $7. proc_1-proc_50;

        * add poa variable based on DX - use N for now *;
        array xdx[50] $7. dx1-dx50;
        array xdiag[50] $7. dx_cce1-dx_cce50;
        array xpoa[50] $1. DX1POA DX2POA DX3POA DX4POA DX5POA
                        DX6POA DX7POA DX8POA DX9POA DX10POA
```

```
                                DX11POA DX12POA DX13POA DX14POA DX15POA
                                DX16POA DX17POA DX18POA DX19POA DX20POA
                                DX21POA DX22POA DX23POA DX24POA DX25POA
                                DX26POA DX27POA DX28POA DX29POA DX30POA
                                DX31POA DX32POA DX33POA DX34POA DX35POA
                                DX36POA DX37POA DX38POA DX39POA DX40POA
                                DX41POA DX42POA DX43POA DX44POA DX45POA
                                DX46POA DX47POA DX48POA DX49POA DX50POA;
          do j = 1 to 50;
                    * 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, G.DX21POA , G.DX22POA , G.DX23POA , G.DX24POA ,
         G.DX25POA, G.DX26POA , G.DX27POA , G.DX28POA , G.DX29POA , G.DX30POA, G.DX31POA ,
         G.DX32POA , G.DX33POA , G.DX34POA , G.DX35POA, G.DX36POA , G.DX37POA , G.DX38POA ,
         G.DX39POA , G.DX40POA, G.DX41POA , G.DX42POA , G.DX43POA , G.DX44POA , G.DX45POA,
         G.DX46POA , G.DX47POA , G.DX48POA , G.DX49POA , G.DX50POA

 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

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	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
O00-09A	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.

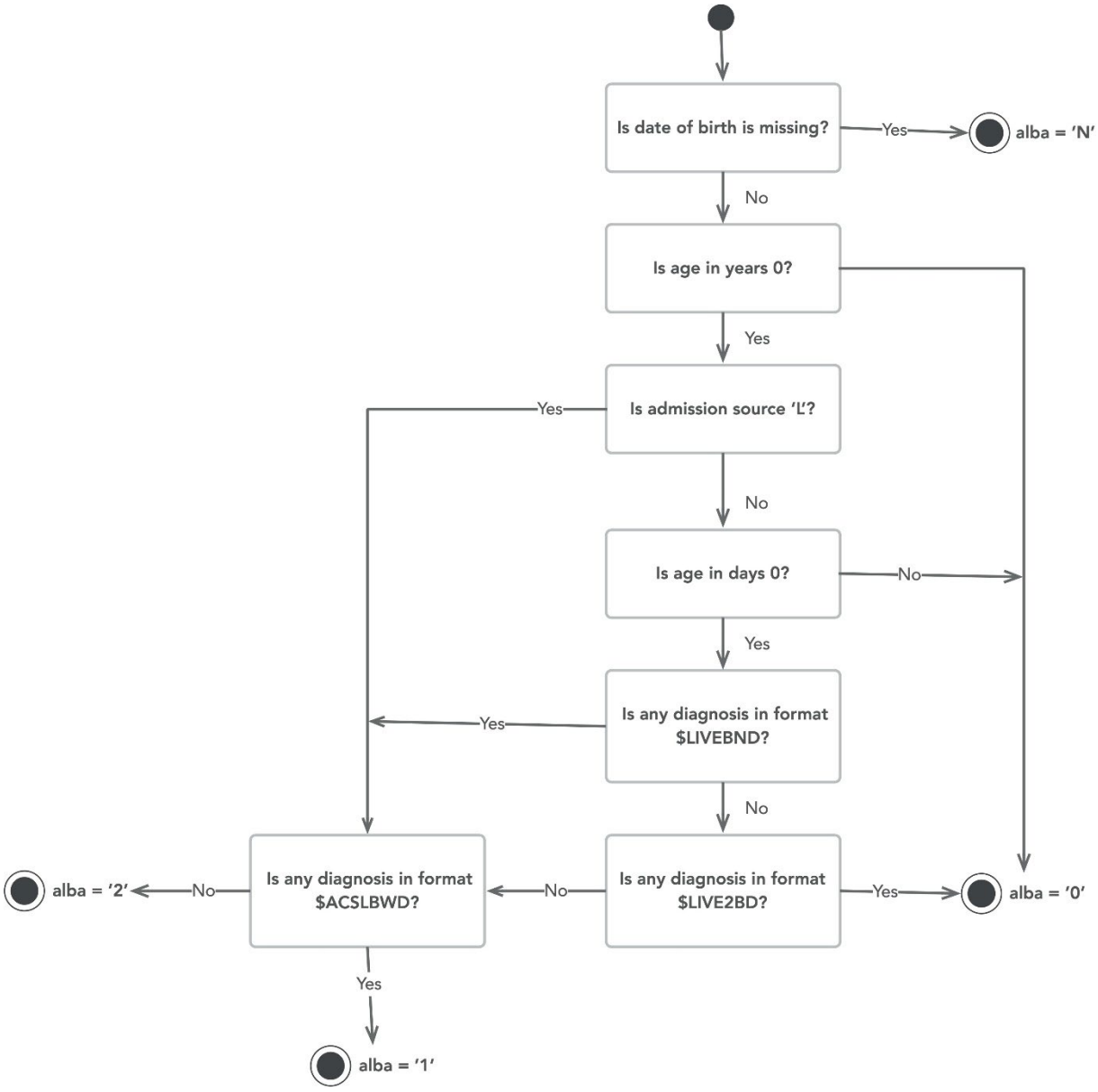
If the ENCOUNTER_TYPE_EH is 'Preadmit,' 'Prereg,' 'Between Visit' or 'Referral Tracking' then make the start and end dates for that segment the same.

APPENDIX F. 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.



APPENDIX G. Present on Admission (POA) Fix

MHS GENESIS was not able to accept Present on Admission (POA) data/flag HL7 message transmitted from Coding and Compliance Editor (CCE) prior to Aug 23, 2021.

MHS GENESIS and other downstream systems did not have the required fields to calculate accurate MS-DRGs due to the missing POA Flags. Resulting in ungroupable or inaccurate calculations.

The fix was applied by using a CCE extract at the FIN+Diagnosis level. The order of the diagnosis codes was maintained and just the POA values from the CCE extract file were applied in processing.

APPENDIX H. Coding Compliance Editor (CCE) Fix

The CCE Fix File is in the MDR in the following location:

- /mdr/genesis/int/tmci/cce_fix/cce_fix_file.sas7bdat

1. There are two problems that are being addressed with this data:
    a. POA Indicators. For the ten impacted MTFs, the POA indicators in Millennium (and thus the MDR) are incorrect and are shifted by one. These POA indicators with be overwritten with the POA indicators from CCE, which are correct. This is very similar to the Original POA Fix File, only in that case, there was no data in the MDR, and it was being sourced from CCE.
    b. Stuck records. For all the MTFs in this file, there could be "stuck" records. These are records that were coded in CCE, but the codes never flowed from CCE into Millennium (and thus the MDR) and they erroneously appear as inferred records in our data. The goal is to bring those codes in from CCE and put them on the MDR data.

2. CCE Fix File:
    a. The CCE Fix File includes additional MTFs (Pendleton, Lewis, San Diego, Leonard, Irwin, Carson, Nellis, Riley, Travis). These MTFs are expected to have minimal impact.
    b. The CCE Fix File (in theory) goes back to IOC (7/14/17, when Oak Harbor went live), where my ad hoc process only went back to FY20. The older FYs (FY17-FY19) are expected to have minimal impact.

3. Comparing the two sources (MDR and CCE) there is a bit of a Venn diagram:
    a. Record is in MDR only. If there is no data in CCE for that record, just leave it alone most of the time.
    b. Test whether the order of codes in CCE matches the order of codes in the raw Millennium Diagnosis table. If the order is maintained, would result in "un-shift" the POA indicators coming from Millennium.
    c. If the record was from one of the MTFs impacted by the POA problem and the data didn't have CCE data to fix it, remove all the codes on the record and called it inferred.

4. Record is in both MDR and CCE.
    a. Record is coded in both MDR and CCE.
        i. If the record is from one of the MTFs impacted by the POA problem and in the impacted time frame (coded in CCE), overwrite the POA indicators in MDR with those from CCE.
        ii. If the record is not from one of the impacted MTFs, recommend deleting all the codes from the MDR data and replacing them with the codes from CCE.
        iii. Use the order of diagnosis codes as they appear in CCE.
    b. Record is inferred (not coded) in MDR but coded in CCE. In this case, bring in all the codes (diagnosis, POA, procedure, procedure dates) from CCE and added them to the MDR file (and changed adm_infr_flag to '0').

5. If the record in MDR indicates that it was coded in 3M360 (contributor_system_dx = '3M CODING AND REIMBURSEMENT'), no action.

6. Note that some of this merging must happen at the FIN/encounter level and some must happen at the FIN + Diagnosis Code level.

a. For example, when fixing POA indicators, the order of the diagnosis codes does not match between MDR and CCE. The match based on the diagnosis code itself (e.g. F840), not on its position in the array. This is the same as how the Original POA Fix File was applied. When dealing with stuck records, you can merge at the FIN level (check that MDR is inferred, bring in the entire array of codes from CCE).

7. After all the codes are fixed/added (diagnosis, POA, procedure), the data needs to be regrouped to get the MS-DRG.