NEMSIS TAC Whitepaper

NEMSIS V3 External Standards Maintenance

Date

June 24, 2020 (Draft)
June 25, 2021 (Corrected RxNorm relationship mappings)
September 23, 2021 (ICD-10-CM URL changes for FY2022)

Authors

Joshua Legler – NEMSIS Consultant

Introduction

This document provides technical recommendations for maintaining the following externally sourced data sets in NEMSIS-compliant software:

- ICD-10-CM diagnosis
- RxNorm medication
- SNOMED CT procedure

The following externally sourced data sets are *not* addressed in this document:

- ANSI county
- ANSI state
- Census American Community Survey data
- Census decennial data
- Census regions and divisions
- Census tract
- Census ZCTA relationship
- GNIS City
- ICD-10-PCS procedure
- ISO country
- NASEMSO regions
- NHTSA regions
- NPI provider
- ZIP code

ICD-10-CM Diagnosis

ICD-10-CM is a US modification of the international ICD-10 code set for diagnoses, used in the NEMSIS standard for recording symptoms, impressions, causes of injury, medical history, and hospital diagnoses. The following guidance focuses on leveraging the hierarchical nature of ICD-10 to provide roll-ups for analysis.

Data source: US Centers for Disease Control and Prevention (CDC) National Center for Health Statistics (NCHS), under authorization from the World Health Organization (WHO)

Update frequency: Annual (typically June–August, occasional addenda)

Releases site: https://www.cdc.gov/nchs/icd/icd10cm.htm

Documentation: https://ftp.cdc.gov/pub/Health_Statistics/NCHS/Publications/ICD10CM/2022/10cmguidelines-FY2022-7-2022-7-15-21-update-508.pdf

Source file:

- **FY2022 and later:** https://ftp.cdc.gov/pub/Health_Statistics/NCHS/Publications/ICD10CM/{YYYY}/Table and Index zip.zip/Table and Index/icd10cm-tabular-{YYYY}.xml
- **FY2021 and earlier:** ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Publications/ICD10CM/{YYYY}/icd10cm_tabular_{YYYY}.xml

Hierarchy:

Level	Example Code	Example Description
Chapter	7	Diseases of the eye and adnexa
Section	H53-H54	Visual disturbances and blindness
Category	H54	Blindness and low vision
Subcategory1	H54.5	Low vision, one eye
Subcategory2	H54.52	Low vision, left eye, normal vision right eye
Subcategory3	H54.52A	Low vision, left eye, category 1-2
Diagnosis	H54.52A2	Low vision left eye category 2, normal vision right eye

Suggested Mapping:

Destination Column	Data Type	Source xPath
DiagnosisCodeKey	INTEGER	(Autoincrement)
DiagnosisCodeType	VARCHAR(8)	“ICD10CM”
DiagnosisCode	VARCHAR(8)	diag/name
DiagnosisCodeDescr	VARCHAR(255)	diag/desc
DiagnosisChapterCode	TINYINT	diag/ancestor::chapter/name
DiagnosisChapterDescr	VARCHAR(255)	diag/ancestor::chapter/desc
DiagnosisSectionCode	VARCHAR(7)	diag/ancestor::section/@id
DiagnosisSectionDescr	VARCHAR(255)	diag/ancestor::section/desc

DiagnosisCategoryCode	CHAR(3)	diag/ancestor-or-self::diag[count(ancestor::diag) = 0]/name
DiagnosisCategoryDescr	VARCHAR(255)	diag/ancestor-or-self::diag[count(ancestor::diag) = 0]/desc
DiagnosisSubcategory1Code	VARCHAR(8)	diag/(ancestor-or-self::diag [count(ancestor::diag) = 1], .)[1]/name
DiagnosisSubcategory1Descr	VARCHAR(255)	diag/(ancestor-or-self::diag [count(ancestor::diag) = 1], .)[1]/desc
DiagnosisSubcategory2Code	VARCHAR(8)	diag/(ancestor-or-self::diag [count(ancestor::diag) = 2], .)[1]/name
DiagnosisSubcategory2Descr	VARCHAR(255)	diag/(ancestor-or-self::diag [count(ancestor::diag) = 2], .)[1]/desc
DiagnosisSubcategory3Code	VARCHAR(8)	diag/(ancestor-or-self::diag [count(ancestor::diag) = 3], .)[1]/name
DiagnosisSubcategory4Descr	VARCHAR(255)	diag/(ancestor-or-self::diag [count(ancestor::diag) = 3], .)[1]/desc
active	BOOLEAN	true()

The business key is DiagnosisCode.

Additional mapping for "7th Characters:"

The source file does not explicitly enumerate all possible ICD-10-CM codes. It uses a <sevenChrDef> element to enumerate a set of characters and description texts that should be appended to certain codes, resulting in additional codes. For example, E08.35 has a <sevenChrDef> that enumerates four different suffixes to be appended to diagnoses that are children of E08.35:

```
<sevenChrDef>
  <extension char="1">right eye</extension>
  <extension char="2">left eye</extension>
  <extension char="3">bilateral</extension>
  <extension char="9">unspecified eye</extension>
</sevenChrDef>
```

For each child diagnosis, the table should contain a row representing that child (as processed using the basic mapping). In addition to that row, there should be a row for each sevenChrDef extension, where all columns are the same except for DiagnosisCodeKey, DiagnosisCode, and DiagnosisCodeDescr, for example:

DiagnosisCode	**DiagnosisCodeDescr**
E08.351	Diabetes mellitus due to underlying condition with proliferative diabetic retinopathy with macular edema
E08.351**1**	Diabetes mellitus due to underlying condition with proliferative diabetic retinopathy with macular edema**, right eye**
E08.351**2**	Diabetes mellitus due to underlying condition with proliferative diabetic retinopathy with macular edema**, left eye**

E08.351**3**	Diabetes mellitus due to underlying condition with proliferative diabetic retinopathy with macular edema**, bilateral eye**
E08.351**9**	Diabetes mellitus due to underlying condition with proliferative diabetic retinopathy with macular edema**, unspecified eye**

If a diagnosis has a SevenChrDef but no child diagnoses, the SevenChrDef extensions should be appended to the diagnosis itself.

If a code to which a SevenChrDef extension is being appended is less than eight characters long including the dot, it should be padded with "X", for example:

DiagnosisCode	**DiagnosisCodeDescr**
M48.40	Fatigue fracture of vertebra, site unspecified
M48.40**XA**	Fatigue fracture of vertebra, site unspecified**, initial encounter for fracture**
Etc.	

The hierarchy below the Section level is ragged: a code may be at the Category, Subcategory1, Subcategory2, Subcategory3, or Diagnosis level. The hierarchy should be "right-filled". For example, "H54" is a diagnosis code. It is also a category code, with no subcategories. Category, Subcategory1, Subcategory2, and Subcategory3 should all be filled with "H54" and its accompanying description.

Chapter Code	**Section Code**	**Category Code**	**Subcategory 1 Code**	**Subcategory 2 Code**	**Subcategory 3 Code**	**Diagnosis Code**
7	H53-H54	H54	*H54*	*H54*	*H54*	*H54*
7	H53-H54	H54	H54.0	*H54.0*	*H54.0*	*H54.0*
7	H53-H54	H54	H54.0	H54.0X	*H54.0X*	*H54.0X*
7	H53-H54	H54	H54.0	H54.0X	H54.0X3	*H54.0X3*
7	H53-H54	H54	H54.0	H54.0X	H54.0X3	H54.0X33

All descriptions in the table should be updated to the descriptions in the data source.

Any codes with a DiagnosisCodeType of "ICD10CM" that exist in the table but not in the data source should be kept in the table, and the active flag should be set to false. This allows historically consistent reporting on codes that have been discontinued but were used in the past (or even may still be in use by systems that have not appropriately updated).

Any codes that exist in the data source but not in the table should be inserted into the table.

RxNorm Medication

RxNorm is a US standard used in NEMSIS for medications. The standard includes information about medications from many perspectives, including ingredients, dose forms, brand names, and others, with relationships defined between them. The following guidance focuses on leveraging the relationships to map back to the ingredients used. This enables analysis, for example, of the usage of naloxone regardless of whether it was reported as naloxone, naloxone hydrochloride, Narcan, naloxone nasal product, naloxone hydrochloride 40 MG/ML, or several other codes.

Data source: US National Institutes of Health (NIH) National Library of Medicine (NLM)

Update frequency: monthly (first Monday; the following day if Monday is a federal holiday; weekly incremental releases are also available)

Releases site: https://www.nlm.nih.gov/research/umls/rxnorm/docs/rxnormfiles.html

Documentation: https://www.nlm.nih.gov/research/umls/rxnorm/docs/index.html

Source files: https://download.nlm.nih.gov/umls/kss/rxnorm/RxNorm_full_current.zip or https://download.nlm.nih.gov/umls/kss/rxnorm/RxNorm_full_MMDDYYYY.zip (UMLS login required; downloads may be automated using https://download.nlm.nih.gov/rxnorm/terminology_download_script.zip)

Staging Tables: Load RXNCONSO.RRF and RXNREL.RRF into staging tables. Scripts are provided for MySQL and Oracle.

Suggested Mapping:

Destination Column	DDL Changes	Source Column (RXNCONSO.RRF)
MedicationCodeKey	INTEGER	(Autoincrement)
MedicationCodeType	VARCHAR(8)	"RXNORM"
MedicationCodeId	VARCHAR(8)	RXAUI
MedicationCodeTermType	VARCHAR(20)	TTY
MedicationCode	VARCHAR(8)	RXCUI
MedicationCodeDescr	VARCHAR(3000)	STR
MedicationCodeIngredients	VARCHAR(3000)	STR (see below for additional ETL)
active	BOOLEAN	true()

For inserting/updating, the business key is MedicationCodeId. (For processing PCR data into the data warehouse, the business key is MedicationCode.)

RXNCONSO.RRF can contain multiple rows with the same RXCUI code. The goal is to have one row for each RXCUI code in DimDiagnosisCode. Use the following algorithm:

1. Filter to rows with SAB= "RXNORM"
2. Exclude rows with TTY = "PSN", "SY", or "TMSY". All such rows are synonyms of other rows.

3. Insert/update DimDiagnosisCode.

All descriptions in the database table should be updated to the descriptions in the data source.

Any codes with a MedicationCodeType of “RXNORM” that exist in the database table but not in the data source should be kept in the table, and the active flag should be set to false. This allows historically consistent reporting on codes that have been discontinued but were used in the past (or even may still be in use by systems that have not appropriately updated).

Any codes that exist in the staging table after the above processing steps but not in the database table should be inserted into the database table.

Additional mapping for ingredients roll-up:

All term types can be mapped to “multiple ingredient” (MIN) or “ingredient” (IN) codes using the relationships defined in RXNREL.RRF. The mapping paths are as follows:

TTY	Name	Path to MIN/IN					
BN	Brand Name					tradename_of	IN
BN	Brand Name	ingredient_of	SBD	tradename_of	SCD	has_ingredients	MIN
BPCK	Brand Name Pack	contains	SBD	consists_of	SCDC	has_ingredient	IN
BPCK	Brand Name Pack	contains	SBD	tradename_of	SCD	has_ingredients	MIN
BPCK	Brand Name Pack	contains	SCD	has_ingredient	BN	has_ingredient	IN
BPCK	Brand Name Pack			contains	SCD	has_ingredients	MIN
DF	Dose Form	*ignore*					
DFG	Dose Form Group	*ignore*					
GPCK	Generic Pack	contains	SCD	consists_of	SCDC	has_ingredient	IN
GPCK	Generic Pack			contains	SCD	has_ingredients	MIN
IN	Ingredient	.					.
MIN	Multiple Ingredients	.					.
PIN	Precise Ingredient					form_of	IN
PSN	Prescribable Name	*ignore*					
SBD	Semantic Branded Drug			consists_of	SCDC	has_ingredient	IN
SBD	Semantic Branded Drug			tradename_of	SCD	has_ingredients	MIN
SBDC	Semantic Branded Drug Component			tradename_of	SCDC	has_ingredient	IN
SBDC	Semantic Branded Drug Component	constitutes	SBD	tradename_of	SCD	has_ingredients	MIN
SBDF	Semantic Branded Drug Form			tradename_of	SCDF	has_ingredient	IN
SBDF	Semantic Branded Drug Form	inverse_isa	SBD	tradename_of	SCD	has_ingredients	MIN
SBDG	Semantic Branded Dose Form Group			has_ingredient	BN	tradename_of	IN
SBDG	Semantic Branded Dose Form Group	inverse_isa	SBD	tradename_of	SCD	has_ingredients	MIN
SCD	Semantic Clinical Drug			consists_of	SCDC	has_ingredient	IN
SCD	Semantic Clinical Drug					has_ingredients	MIN

SCDC	Semantic Clinical Drug Component			has_ingredient IN
SCDF	Semantic Clinical Drug Form			has_ingredient IN
SCDF	Semantic Clinical Drug Form		inverse_isa SCD	has_ingredients MIN
SCDG	Semantic Clinical Dose Form Group			has_ingredient IN
SCDG	Semantic Clinical Dose Form Group		inverse_isa SCD	has_ingredients MIN
SY	Synonym	*ignore*		
TMSY	Tall Man Lettering Synonym	*ignore*		

For codes with TTY = “MIN” or “IN”, MedicationCodeIngredients will be the same as MedicationCodeDescr, so no further action is needed.

For codes of other TTYs, follow the mapping paths detailed above. Some mappings will lead to a MIN code as well as multiple IN codes. The MIN code should be used when present.

Here is an example query to map from SCD codes to IN codes:

```
SELECT TOP 1 C3.RXCUI, C3.STR
FROM RXNCONSO C1
JOIN RXNREL R1 ON C1.RXCUI  = R1.RXCUI2 AND R1.RELA = "consists_of"
JOIN RXNCONSO C2 ON R1.RXCUI1 = C2.RXCUI
JOIN RXNREL R2 ONC2.RXCUI = R2.RXCUI2 AND R2.RELA = "has_ingredient"
JOIN RXNCONSO C3 ON R2.RXCUI1 = C3.RXCUI
WHERE C1.TTY = "SCD" AND C2.TTY = "SCDC" AND C3.TTY = "IN"
ORDER BY C3.TTY DESC
```

Use the results of the mapping queries to map C2.STR to the MedicationCodeIngredients column in the database table.

SNOMED CT Procedure

SNOMED-CT US Edition is a US release of the International SNOMED-CT data set for clinical terms, used in the NEMSIS standard for recording EMS procedures. Procedures account for about 10% of the SNOMED data set.

SNOMED is hierarchical; however, it is a multiple-inheritance hierarchy, where a concept can have multiple parent concepts. For example, the "Abdominal thrust" procedure has two parents: "Airway procedure" and "Procedure on abdomen".

Data source: US National Institutes of Health (NIH) National Library of Medicine (NLM), under the authorization from SNOMED International, aka International Health Terminology Standards Development Organisation (IHTSDO)

Update frequency: Semi-annual (March and September)

Releases site: https://www.nlm.nih.gov/healthit/snomedct/us_edition.html

Documentation: https://confluence.ihtsdotools.org/display/DOCRELFMT/SNOMED+CT+Release+File+Specifications

Source file:
https://download.nlm.nih.gov/mlb/utsauth/USExt/SnomedCT_USEditionRF2_PRODUCTION_{YYYYMMDD}T120000Z.zip/SnomedCT_USEditionRF2_PRODUCTION_{YYYYMMDD}T120000Z/Snapshot/Terminology/sct2_Description_Snapshot-en_US1000124_{YYMMDD}.txt (UMLS login required; downloads may be automated using https://download.nlm.nih.gov/rxnorm/terminology_download_script.zip)

Suggested mapping:

The NEMSIS Suggested List documentation recommends only using SNOMED codes with a semantic tag of "procedure" or "regime/therapy". However, implementations have used SNOMED codes with other semantic tags, such as "equipment."

The source file contains multiple rows for each concept code. One row contains the active fully specified name, while other rows contain synonyms or inactive entries. Prior to loading into a database table, apply the following filter:

active = 1 -- (active)
AND typeId = 900000000000003001 -- (fully specified name)

Many but not all values in the "term" column of the source table contain a semantic tag at the end. The semantic tags are:

Semantic tag
(administration method)
(assessment scale)

Semantic tag
(attribute)
(basic dose form)

Semantic tag
(body structure)
(cell structure)

Semantic tag
(cell)
(clinical drug)
(core metadata concept)
(disorder)
(disposition)
(dose form)
(environment)
(ethnic group)
(event)
(finding)
(foundation metadata concept)
(geographic location)
(inactive concept)
(intended site)
(life style)
(link assertion)

Semantic tag
(linkage concept)
(medicinal product form)
(medicinal product)
(morphologic abnormality)
(namespace concept)
(navigational concept)
(number)
(observable entity)
(occupation)
(organism)
(OWL metadata concept)
(person)
(physical force)
(physical object)
(procedure)
(product name)
(product)

Semantic tag
(qualifier value)
(racial group)
(record artifact)
(regime/therapy)
(release characteristic)
(religion/philosophy)
(role)
(situation)
(social concept)
(specimen)
(staging scale)
(state of matter)
(substance)
(supplier)
(transformation)
(tumor staging)
(unit of presentation)

Split the semantic tag off the end of each text value, remove the "(" and ")" and capitalize the first letter. Remove the extra whitespace remaining at the end of the text value. For values with no semantic tag, set the semantic tag to "None".

Destination Column	DDL Changes	Source Column (filtered sct2_Description….txt)
ProcedureCodeKey	INTEGER	(Autoincrement)
ProcedureCodeType	VARCHAR(8)	"SNOMED"
ProcedureCode	VARCHAR(18)	conceptId
ProcedureCodeDescr	VARCHAR(3000)	term (after ETL modification described above)
ProcedureCodeSemanticType	VARCHAR(30)	semantic tag from term (after ETL modification described above)
active	BOOLEAN	true()

All descriptions in the table should be updated to the descriptions in the data source (after ETL modifications described above).

Any codes with a ProcedureCodeType of "SNOMED" that exist in the table but not in the data source should be kept in the table, and the active flag should be set to false. This allows historically consistent reporting on codes that have been discontinued but were used in the past (or even may still be in use by systems that have not appropriately updated).

Any codes that exist in the data source but not in the table should be inserted into the table.