OMOP Common Data Model Specifications

Christian Reich, Patrick Ryan, Rimma Belenkaya, Karthik Natarajan and Clair Blacketer

2019-02-06

Contents

1 License

2 Background

The Observational Medical Outcomes Partnership (OMOP) was a public-private partnership established to inform the appropriate use of observational healthcare databases for studying the effects of medical products. Over the course of the 5-year project and through its community of researchers from industry, government, and academia, OMOP successfully achieved its aims to:

- Conduct methodological research to empirically evaluate the performance of various analytical methods on their ability to identify true associations and avoid false findings
- Develop tools and capabilities for transforming, characterizing, and analysing disparate data sources across the health care delivery spectrum
- Establish a shared resource so that the broader research community can collaboratively advance the science

The results of OMOP's research has been widely published and presented at scientific conferences, including annual symposia.

The OMOP Legacy continues…

The community is actively using the OMOP Common Data Model for their various research purposes. Those tools will continue to be maintained and supported, and information about this work is available in the public domain.

The Observational Health Data Sciences and Informatics (OHDSI) has been established as a multi-stakeholder, interdisciplinary collaborative to create open-source solutions that bring out the value of observational health data through large-scale analytics. The OHDSI collaborative includes all of the original OMOP research investigators, and will develop its tools using the OMOP Common Data Model. Learn more at ohdsi.org.

The OMOP Common Data Model will continue to be an open-source community standard for observational healthcare data. The model specifications and associated work products will be placed in the public domain, and the entire research community is encouraged to use these tools to support everybody's own research activities.

2.1 The Role of the Common Data Model

No single observational data source provides a comprehensive view of the clinical data a patient accumulates while receiving healthcare, and therefore none can be sufficient to meet all expected outcome analysis needs. This explains the need for assessing and analyzing multiple data sources concurrently using a common data standard. This standard is provided by the OMOP Common Data Model (CDM).

The CDM is designed to support the conduct of research to identify and evaluate associations between interventions (drug exposure, procedures, healthcare policy changes etc.) and outcomes caused by these interventions (condition occurrences, procedures, drug exposure etc.). Outcomes can be efficacious (benefit) or adverse (safety risk). Often times, specific patient cohorts (e.g., those taking a certain drug or suffering from a certain disease) may be defined for treatments or outcomes, using clinical events (diagnoses, observations, procedures, etc.) that occur in predefined temporal relationships to each other. The CDM, combined with its standardized content (via the Standardized Vocabularies), will ensure that research methods can be systematically applied to produce meaningfully comparable and reproducible results.

2.2 Design Principles

The CDM is designed to include all observational health data elements (experiences of the patient receiving health care) that are relevant for analysis use cases to support the generation of reliable scientific evidence about disease natural history, healthcare delivery, effects of medical interventions, the identification of demographic information, health care interventions and outcomes.

Therefore, the CDM is designed to store observational data to allow for research, under the following principles:

- **Suitability for purpose:** The CDM aims to provide data organized in a way optimal for analysis, rather than for the purpose of addressing the operational needs of health care providers or payers.
- **Data protection:** All data that might jeopardize the identity and protection of patients, such as names, precise birthdays etc. are limited. Exceptions are possible where the research expressly requires more detailed information, such as precise birth dates for the study of infants.
- **Design of domains:** The domains are modeled in a person-centric relational data model, where for each record the identity of the person and a date is captured as a minimum.
- **Rationale for domains:** Domains are identified and separately defined in an entity-relationship model if they have an analysis use case and the domain has specific attributes that are not otherwise applicable. All other data can be preserved as an observation in an entity-attribute-value structure.
- **Standardized Vocabularies:** To standardize the content of those records, the CDM relies on the Standardized Vocabularies containing all necessary and appropriate corresponding standard healthcare concepts.
- **Reuse of existing vocabularies:** If possible, these concepts are leveraged from national or industry standardization or vocabulary definition organizations or initiatives, such as the National Library of Medicine, the Department of Veterans' Affairs, the Center of Disease Control and Prevention, etc.
- **Maintaining source codes:** Even though all codes are mapped to the Standardized Vocabularies, the model also stores the original source code to ensure no information is lost.
- **Technology neutrality:** The CDM does not require a specific technology. It can be realized in any relational database, such as Oracle, SQL Server etc., or as SAS analytical datasets.
- **Scalability:** The CDM is optimized for data processing and computational analysis to accommodate data sources that vary in size, including databases with up to hundreds of millions of persons and billions of clinical observations.
- **Backwards compatibility:** All changes from previous CDMs are clearly delineated in the github repository. Older versions of the CDM can be easily created from the CDMv5, and no information is lost that was present previously.

2.3 Data Model Conventions

There are a number of implicit and explicit conventions that have been adopted in the CDM. Developers of methods that run against the CDM need to understand these conventions.

2.3.1 General conventions of schemas

New to CDM v6.0 is the concept of schemas. This allows for more separation between read-only and writeable tables. The clinical data, event, and vocabulary tables are in the 'CDM' schema and tables that need to be manipulated by web-based tools or end users have moved to the 'Results' schema. Currently the only two tables in the 'Results' schema are COHORT and COHORT_DEFINITON, though likely more will be added over the course of v6.0 point releases.

2.3.2 General conventions of data tables

The CDM is platform-independent. Data types are defined generically using ANSI SQL data types (VARCHAR, INTEGER, FLOAT, DATE, DATETIME, CLOB). Precision is provided only for VARCHAR. It reflects the minimal required string length and can be expanded within a CDM instantiation. The CDM does not prescribe the date and datetime format. Standard queries against CDM may vary for local instantiations and date/datetime configurations.

In most cases, the first field in each table ends in '_ID', containing a record identifier that can be used as a foreign key in another table.

2.3.3 General conventions of fields

Variable names across all tables follow one convention:

Notation	Description
_SOURCE_VALUE	Verbatim information from the source data, typically used in ETL to map to CONCEPT_ID, and not to be used by any standard analytics. For example, CONDITION_SOURCE_VALUE = '787.02' was the ICD-9 code captured as a diagnosis from the administrative claim.
_ID	Unique identifiers for key entities, which can serve as foreign keys to establish relationships across entities. For example, PERSON_ID uniquely identifies each individual. VISIT_OCCURRENCE_ID uniquely identifies a PERSON encounter at a point of care.
_CONCEPT_ID	Foreign key into the Standardized Vocabularies (i.e. the standard_concept attribute for the corresponding term is true), which serves as the primary basis for all standardized analytics. For example, CONDITION_CONCEPT_ID = 31967 contains the reference value for the SNOMED concept of 'Nausea'
_SOURCE_CONCEPT_ID	Foreign key into the Standardized Vocabularies representing the concept and terminology used in the source data, when applicable. For example, CONDITION_SOURCE_CONCEPT_ID = 45431665 denotes the concept of 'Nausea' in the Read terminology; the analogous CONDITION_CONCEPT_ID might be 31967, since SNOMED-CT is the Standardized Vocabulary for most clinical diagnoses and findings.
_TYPE_CONCEPT_ID	Delineates the origin of the source information, standardized within the Standardized Vocabularies. For example, DRUG_TYPE_CONCEPT_ID can allow analysts to discriminate between 'Pharmacy dispensing' and 'Prescription written'

2.3.4 Representation of content through Concepts

In CDM data tables the meaning of the content of each record is represented using Concepts. Concepts are stored with their CONCEPT_ID as foreign keys to the CONCEPT table in the Standardized Vocabularies, which contains Concepts necessary to describe the healthcare experience of a patient. If a Standard Concept does not exist or cannot be identified, the Concept with the CONCEPT_ID 0 is used, representing a non-existing or un-mappable concept.

Records in the CONCEPT table contain all the detailed information about it (name, domain, class etc.). Concepts, Concept Relationships and other information relating to Concepts is contained in the tables of the Standardized Vocabularies.

2.3.5 Difference between Concept IDs and Source Values

Many tables contain equivalent information multiple times: As a Source Value, a Source Concept and as a Standard Concept.

- Source Values contain the codes from public code systems such as ICD-9-CM, NDC, CPT-4 etc. or locally controlled vocabularies (such as F for female and M for male) copied from the source data. Source Values are stored in the *_SOURCE_VALUE fields in the data tables.
- Concepts are CDM-specific entities that represent the meaning of a clinical fact. Most concepts are based on code systems used in healthcare (called Source Concepts), while others were created de-novo (CONCEPT_CODE = 'OMOP generated'). Concepts have unique IDs across all domains.
- Source Concepts are the concepts that represent the code used in the source. Source Concepts are only used for common healthcare code systems, not for OMOP-generated Concepts. Source Concepts are stored in the *_SOURCE_CONCEPT_ID field in the data tables.
- Standard Concepts are those concepts that are used to define the unique meaning of a clinical entity. For each entity there is one Standard Concept. Standard Concepts are typically drawn from existing public vocabulary sources. Concepts that have the equivalent meaning to a Standard Concept are mapped to the Standard Concept. Standard Concepts are referred to in the _CONCEPT_ID field of the data tables.

Source Values are only provided for convenience and quality assurance (QA) purposes. Source Values and Source Concepts are optional, while Standard Concepts are mandatory. Source Values may contain information that is only meaningful in the context of a specific data source.

2.3.6 Difference between general Concepts and Type Concepts

Type Concepts (ending in _TYPE_CONCEPT_ID) and general Concepts (ending in _CONCEPT_ID) are part of many tables. The former are special Concepts with the purpose of indicating where the data are derived from in the source. For example, the Type Concept field can be used to distinguish a DRUG_EXPOSURE record that is derived from a pharmacy-dispensing claim from one indicative of a prescription written in an electronic health record (EHR).

2.3.7 Time span of available data

Data tables for clinical data contain a datetime stamp (ending in _DATETIME, _START_DATETIME or _END_DATETIME), indicating when that clinical event occurred. As a rule, no record can be outside of a valid OBSERVATION_PERIOD time period. Clinical information that relates to events that happened prior to the first OBSERVATION_PERIOD will be captured as a record in the OBSERVATION table as 'Medical history' (CONCEPT_ID = 43054928), with the OBSERVATION_DATETIME set to the first OBSERVATION_PERIOD_START_DATE of that patient, and the VALUE_AS_CONCEPT_ID set to the corresponding CONCEPT_ID for the condition/drug/procedure that occurred in the past. No data occurring after the last OBSERVATION_PERIOD_END_DATE can be valid records in the CDM. * When mapping source data to the CDM, if time is unknown the default time of 00:00:00 should be chosen. If a time of 00:00:00 is given in the source data it should be shifted to 00:00:01 (THEMIS issue #10).

2.3.8 Content of each table

For the tables of the main domains of the CDM it is imperative that concepts used are strictly limited to the domain. For example, the CONDITION_OCCURRENCE table contains only information about conditions (diagnoses, signs, symptoms), but no information about procedures. Not all source coding schemes adhere to such rules. For example, ICD-9-CM codes, which contain mostly diagnoses of human disease, also contain information about the status of patients having received a procedure. The ICD-9-CM code

V20.3 'Newborn health supervision' defines a continuous procedure and is therefore stored in the PROCEDURE_OCCURRENCE table.

2.3.9 Differentiating between Source Values, Source Concept Ids, and Standard Concept Ids

Each table contains fields for Source Values, Source Concept Ids, and Standard Concept Ids.

- Source Values are fields to maintain the verbatim information from the source database, stored as unstructured text, and are generally not to be used by any standardized analytics. There is no standardization for these fields and these columns can be used as the local CDM builders see fit. A typical example would be an ICD-9 code without the decimal from an administrative claim as condition_source_value = '78702' which is how it appeared in the source (THEMIS issue #15).
- Source Concept Ids provide a repeatable representation of the source concept, when the source data are drawn from a commonly-used, internationally-recognized vocabulary that has been distributed with the OMOP Common Data Model. Specific use cases where source vocabulary-specific analytics are required can be accommodated by the use of the *_SOURCE_CONCEPT_ID fields, but these are generally not applicable across disparate data sources. The standard *_CONCEPT_ID fields are **strongly suggested** to be used in all standardized analytics, as specific vocabularies have been established within each data domain to facilitate standardization of both structure and content within the OMOP Common Data Model.

The following provide conventions for processing source data using these three fields in each domain:

When processing data where the source value is either free text or a reference to a coding scheme that is not contained within the Standardized Vocabularies:

- Map all Source Values to the corresponding Standard CONCEPT_IDs. Store the CONCEPT_IDs in the TARGET_CONCEPT_ID field of the SOURCE_TO_CONCEPT_MAP table.
  - If a CONCEPT_ID is not available for the source code, the TARGET_CONCEPT_ID field is set to 0.

When processing your data where Source Value is a reference to a coding scheme contained within the Standardized Vocabularies:

- Find all CONCEPT_IDs in the Source Vocabulary that correspond to your Source Values. Store the result in the SOURCE_CONCEPT_ID field.
  - If the source code follows the same formatting as the distributed vocabulary, the mapping can be directly obtained from the CONCEPT table using the CONCEPT_CODE field.
  - If the source code uses alternative formatting (ex. format has removed decimal point from ICD-9 codes), you will need to perform the formatting transformation within the ETL. In this case, you may wish to store the mappings from original codes to SOURCE_CONCEPT_IDs in the SOURCE_TO_CONCEPT_MAP table.
  - If the source code is not found in a vocabulary, the SOURCE_CONCEPT_ID field is set to 0
- Use the CONCEPT_RELATIONSHIP table to identify the Standard CONCEPT_ID that corresponds to the SOURCE_CONCEPT_ID in the domain.
  - Each SOURCE_CONCEPT_ID can have 1 or more Standard CONCEPT_IDs mapped to it. Each Standard CONCEPT_ID belongs to only one primary domain but when a source CONCEPT_ID maps to multiple Standard CONCEPT_IDs, it is possible for that SOURCE_CONCEPT_ID to result in records being produced across multiple domains. For example, ICD-10-CM code Z34.00 'Encounter for supervision of normal first pregnancy, unspecified trimester' will be mapped to the SNOMED concept 'Routine antenatal care' in the procedure domain and the concept in the condition domain 'Primagravida'. It is also possible for one SOURCE_CONCEPT_ID to map to multiple Standard CONCEPT_IDs within the same domain. For example, ICD-9-CM code 070.43 'Hepatitis E with hepatic coma' maps to the SNOMED concept for 'Acute hepatitis E' and a second SNOMED concept for 'Hepatic coma', in

which case multiple CONDITION_OCCURRENCE records will be generated for the one source value record.
  - If the SOURCE_CONCEPT_ID is not mappable to any Standard CONCEPT_ID, the _CONCEPT_ID field is set to 0.
- Write the data record into the table(s) corresponding to the domain of the Standard CONCEPT_ID(s).
  - If the Source Value has a SOURCE_CONCEPT_ID but the SOURCE_CONCEPT_ID is not mapped to a Standard CONCEPT_ID, then the domain for the data record, and hence it's table location, is determined by the DOMAIN_ID field of the CONCEPT record the SOURCE_CONCEPT_ID refers to. The Standard _CONCEPT_ID is set to 0.
  - If the Source Value cannot be mapped to a SOURCE_CONCEPT_ID or Standard CONCEPT_ID, then direct the data record to the most appropriate CDM domain based on your local knowledge of the intent of the source data and associated value. For example, if the un-mappable Source Value came from a 'diagnosis' table then, in the absence of other information, you may choose to record that fact in the CONDITION_OCCURRENCE table.

Each Standard CONCEPT_ID field has a set of allowable CONCEPT_ID values. The allowable values are defined by the domain of the concepts. For example, there is a domain concept of 'Gender', for which there are only two allowable standard concepts of practical use (8507 - 'Male', 8532- 'Female') and one allowable generic concept to represent a standard notion of 'no information' (concept_id = 0). This 'no information' concept should be used when there is no mapping to a standard concept available or if there is no information available for that field. The exceptions are MEASUREMENT.VALUE_AS_CONCEPT_ID, OBSERVATION.VALUE_AS_CONCEPT_ID, MEASUREMENT.UNIT_CONCEPT_ID, OBSERVATION.UNIT_CONCEPT_ID, MEASUREMENT.OPERATOR_CONCEPT_ID, and OBSERVATION.MODIFIER_CONCEPT_ID, which can be NULL if the data do not contain the information (THEMIS issue #11).

There is no constraint on allowed CONCEPT_IDs within the SOURCE_CONCEPT_ID fields.

2.3.10 Custom SOURCE_TO_CONCEPT_MAPs

When the source data uses coding systems that are not currently in the Standardized Vocabularies (e.g. ICPC codes for diagnoses), the convention is to store the mapping of such source codes to Standard Concepts in the SOURCE_TO_CONCEPT_MAP table. The codes used in the data source can be recorded in the SOURCE_VALUE fields, but no SOURCE_CONCEPT_ID will be available.

Custom source codes are not allowed to map to Standard Concepts that are marked as invalid.

3 Glossary of Terms

Glossary of Terms

Term	Abbr.	Description
Ancestor		The higher level Concept in a hierarchical relationship. Note that ancestors and descendants can be many levels apart from each other.
Average Wholesale Price	AWP	The price manufacturers set for prescription drugs to be purchased at the wholesale level to pharmacies and healthcare provider.
Centers for Disease Control and Prevention	CDC	The Centers for Disease Control and Prevention is a United States federal agency under the Department of Health and Human Services. It works to protect public health and safety by providing information to enhance health decisions.

Term	Abbr.	Description
Common Data Model	CDM	The CDM intends to facilitate observational analyses of disparate healthcare databases. The CDM defines table structures for each of the data entities (e.g., Persons, Visit Occurrence, Drug Exposure, Condition Occurrence, Observation, Procedure Occurrence, etc.). It includes observational data elements that are relevant to identifying exposure to various treatments and defining condition occurrence. The CDM includes both the Standardized Vocabularies of terms and the entity domain tables.
Concept		A concept is the basic unit of information. Concepts may be grouped into a given domain. A concept is a unique term that has a unique and static identifier/name, belongs to a domain, and may exist in relation to other concepts. The vertical relationships consist of "is a" statements that form a logical hierarchy. In general, concepts above a given concept are referred to as ancestors and those below as descendants.
Conceptual Data Model		A conceptual data model is a map of concepts and their relationships. This describes the semantics of an organization and represents a series of assertions about its nature. Specifically, it describes the things of significance to an organization (entity classes), about which it is inclined to collect information, and characteristics of (attributes) and associations between pairs of those things of significance (relationships).
Data mapping		It is the data element mappings between two distinct data models, terminologies, or concepts. Data mapping is the process of creating data element mappings between two distinct data models. Data mapping is used as a first step for a wide variety of data integration tasks.
Demographics		Demographics refer to selected characteristics of persons. Demographics may include data such as race, age, sex, date of birth, location, etc.
Descendant		The lower level Concept in a hierarchical relationship. Note that ancestors and descendants can be many levels apart from each other.
Design Principle		An organized arrangement of one or more elements or principles for a purpose. It identifies core principles and best practices to assist developers to produce software. Thoroughly understanding the goals of stakeholders and designing systems with those goals in mind are the best approaches to successfully deliver results.

Term	Abbr.	Description
Electronic Health Record	EHR	Electronic health record refers to an individual person's medical record in digital format. It may be made up of electronic medical records from many locations and/or sources. The EHR is a longitudinal electronic record of person health information generated by one or more encounters in any care delivery setting. Included in this information are person demographics, progress notes, problems, medications, vital signs, past medical history, immunizations, laboratory data and radiology reports.
Electronic Medical Record	EMR	An electronic medical record is a computerized medical record created in an organization that delivers care, such as a hospital or outpatient setting. Electronic medical records tend to be a part of a local stand-alone health information system that allows storage, retrieval and manipulation of records. This document will reference EHR moving forward even if specific data source might internally use EMR definition.
Extract Transform Load	ETL	Process of getting data out of one data store (Extract), modifying it (Transform), and inserting it into a different data store (Load).
Health Insurance Portability and Accountability Act	HIPAA	A federal law that was designed to allow portability of health insurance between jobs. In addition, it required the creation of a federal law to protect personally identifiable health information; if that did not occur by a specific date (which it did not), HIPAA directed the Department of Health and Human Services (DHHS) to issue federal regulations with the same purpose. DHHS has issued HIPAA privacy regulations (the HIPAA Privacy Rule) as well as other regulations under HIPAA.
Logical Data Model		Logical data models are graphical representation of the business requirements. They describe the things of importance to an organization and how they relate to one another, as well as business definitions and examples. The logical data model can be validated and approved by a business representative, and can be the basis of physical database design.
Primary Care Provider	PCP	A health care provider designated as responsible to provide general medical care to a patient, including evaluation and treatment as well as referral to specialists.
Protected Health Information	PHI	Protected health information under HIPAA includes any individually identifiable health information. Identifiable refers not only to data that is explicitly linked to a particular individual (that's identified information). It also includes health information with data items which reasonably could be expected to allow individual identification. De-identified information is that from which all potentially identifying information has been removed.
Terminology		Technical or special terms used in a business or special subject area.

Term	Abbr.	Description
Vocabulary		A computerized list (as of items of data or words) used for reference (as for information retrieval or word processing).

4 Standardized Vocabularies

CONCEPT
VOCABULARY
DOMAIN
CONCEPT_CLASS
CONCEPT_RELATIONSHIP
RELATIONSHIP
CONCEPT_SYNONYM
CONCEPT_ANCESTOR
SOURCE_TO_CONCEPT_MAP
DRUG_STRENGTH

These tables contain detailed information about the Concepts used in all of the CDM fact tables. The content of the Standardized Vocabularies tables is not generated anew by each CDM implementation. Instead, it is maintained centrally as a service to the community.

A number of assumptions were made for the design of the Standardized Vocabularies tables:

- There is one design which will accommodate all different source terminologies and classifications.
- All terminologies are loaded into the CONCEPT table.
- The key is a newly created concept_id, not the original code of the terminology, because source codes are not unique identifiers across terminologies.
- Some Concepts are declared Standard Concepts, i.e. they are used to represent a certain clinical entity in the data. All Concepts may be Source Concepts; they represent how the entity was coded in the source. Standard Concepts are identified through the standard_concept field in the CONCEPT table.
- Records in the CONCEPT_RELATIONSHIP table define semantic relationships between Concepts. Such relationships can be hierarchical or lateral.
- Records in the CONCEPT_RELATIONSHIP table are used to map Source codes to Standard Concepts, replacing the mechanism of the SOURCE_TO_CONCEPT_MAP table used in prior Standardized Vocabularies versions. The SOURCE_TO_CONCEPT_MAP table is retained as an optional aid to bookkeeping codes not found in the Standardized Vocabularies.
- Chains of hierarchical relationships are recorded in the CONCEPT_ANCESTOR table. Ancestry relationships are only recorded between Standard Concepts that are valid (not deprecated) and are connected through valid and hierarchical relationships in the RELATIONSHIP table (flag DEFINES_ANCESTRY).

The advantage of this approach lies in the preservation of codes and relationships between them without adherence to the multiple different source data structures, a simple design for standardized access, and the optimization of performance for analysis. Navigation among Standard Concepts does not require knowledge of the source vocabulary. Finally, the approach is scalable and future vocabularies can be integrated easily. On the other hand, extensive transformation of source data to the Vocabulary is required and not every source data structure and original source hierarchy can be retained.

Below is an entity-relationship diagram highlighting the tables within the Vocabulary portion of the OMOP Common Data Model:

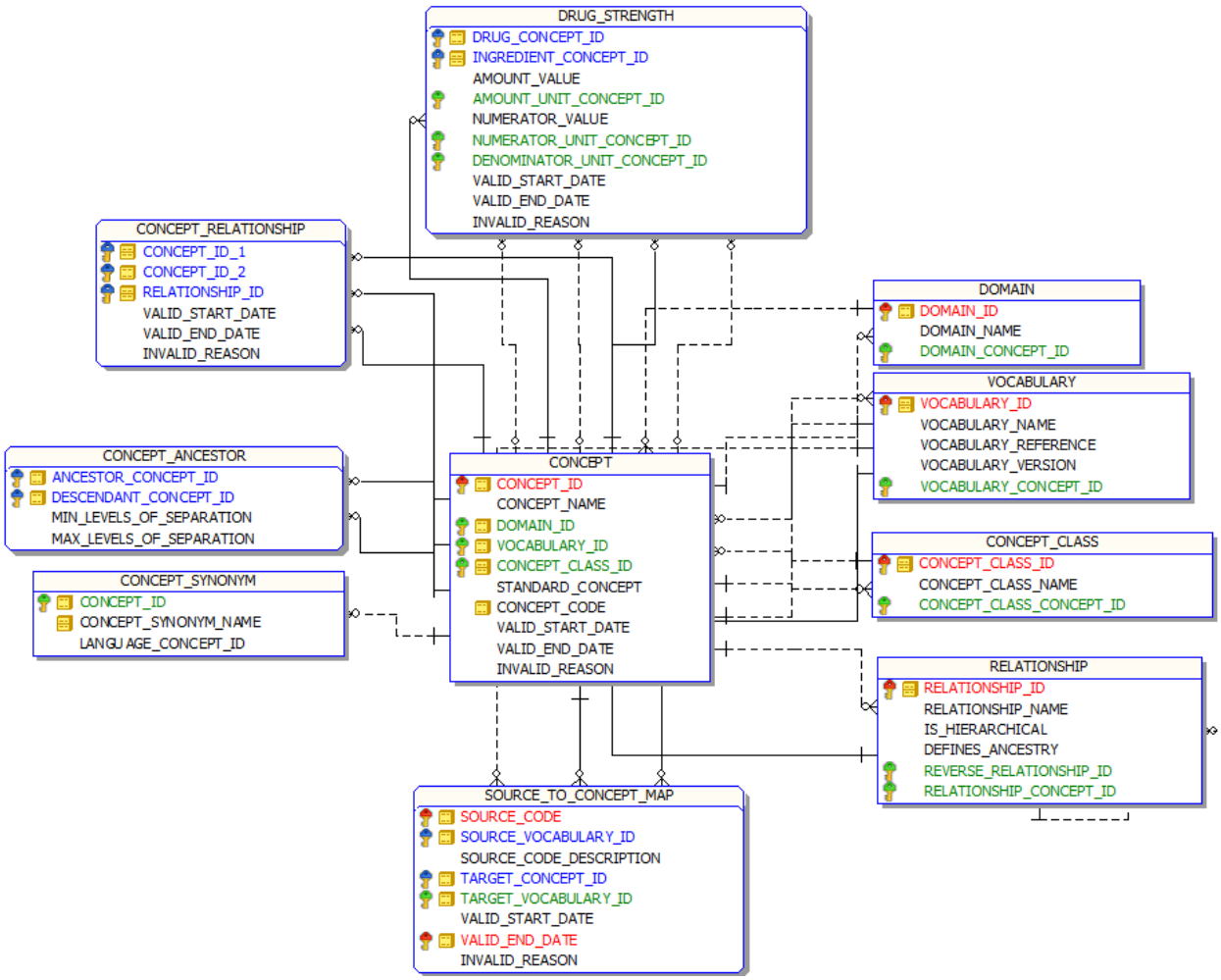

Figure 1: Vocabulary entity-relationship diagram

4.1 CONCEPT

The Standardized Vocabularies contains records, or Concepts, that uniquely identify each fundamental unit of meaning used to express clinical information in all domain tables of the CDM. Concepts are derived from vocabularies, which represent clinical information across a domain (e.g. conditions, drugs, procedures) through the use of codes and associated descriptions. Some Concepts are designated Standard Concepts, meaning these Concepts can be used as normative expressions of a clinical entity within the OMOP Common Data Model and within standardized analytics. Each Standard Concept belongs to one domain, which defines the location where the Concept would be expected to occur within data tables of the CDM.

Concepts can represent broad categories (like 'Cardiovascular disease'), detailed clinical elements ('Myocardial infarction of the anterolateral wall') or modifying characteristics and attributes that define Concepts at various levels of detail (severity of a disease, associated morphology, etc.).

Records in the Standardized Vocabularies tables are derived from national or international vocabularies such as SNOMED-CT, RxNorm, and LOINC, or custom Concepts defined to cover various aspects of observational data analysis. For a detailed description of these vocabularies, their use in the OMOP CDM and their relationships to each other please refer to the specifications.

Field	Required	Type	Description
concept_id	Yes	integer	A unique identifier for each Concept across all domains.
concept_name	Yes	varchar(255)	An unambiguous, meaningful and descriptive name for the Concept.
domain_id	Yes	varchar(20)	A foreign key to the DOMAIN table the Concept belongs to.
vocabulary_id	Yes	varchar(20)	A foreign key to the VOCABULARY table indicating from which source the Concept has been adapted.
concept_class_id	Yes	varchar(20)	The attribute or concept class of the Concept. Examples are 'Clinical Drug', 'Ingredient', 'Clinical Finding' etc.
standard_concept	No	varchar(1)	This flag determines where a Concept is a Standard Concept, i.e. is used in the data, a Classification Concept, or a non-standard Source Concept. The allowables values are 'S' (Standard Concept) and 'C' (Classification Concept), otherwise the content is NULL.
concept_code	Yes	varchar(50)	The concept code represents the identifier of the Concept in the source vocabulary, such as SNOMED-CT concept IDs, RxNorm RXCUIs etc. Note that concept codes are not unique across vocabularies.
valid_start_date	Yes	date	The date when the Concept was first recorded. The default value is 1-Jan-1970, meaning, the Concept has no (known) date of inception.

Field	Required	Type	Description
valid_end_date	Yes	date	The date when the Concept became invalid because it was deleted or superseded (updated) by a new concept. The default value is 31-Dec-2099, meaning, the Concept is valid until it becomes deprecated.
invalid_reason	No	varchar(1)	Reason the Concept was invalidated. Possible values are D (deleted), U (replaced with an update) or NULL when valid_end_date has the default value.

4.1.1 Conventions

Concepts in the Common Data Model are derived from a number of public or proprietary terminologies such as SNOMED-CT and RxNorm, or custom generated to standardize aspects of observational data. Both types of Concepts are integrated based on the following rules:

No.	Convention Description
1	All Concepts are maintained centrally by the CDM and Vocabularies Working Group. Additional concepts can be added, as needed, upon request.
2	For all Concepts, whether they are custom generated or adopted from published terminologies, a unique numeric identifier concept_id is assigned and used as the key to link all observational data to the corresponding Concept reference data.
3	The concept_id of a Concept is persistent, i.e. stays the same for the same Concept between releases of the Standardized Vocabularies.
4	A descriptive name for each Concept is stored as the Concept Name as part of the CONCEPT table. Additional names and descriptions for the Concept are stored as Synonyms in the CONCEPT_SYNONYM table.
5	Each Concept is assigned to a Domain. For Standard Concepts, these is always a single Domain. Source Concepts can be composite or coordinated entities, and therefore can belong to more than one Domain. The domain_id field of the record contains the abbreviation of the Domain, or Domain combination. Please refer to the Standardized Vocabularies specification for details of the Domain Assignment.
6	For details of the Vocabularies adopted for use in the OMOP CDM refer to the Standardized Vocabularies specification.

No.	Convention Description
7	Concept Class designation are attributes of Concepts. Each Vocabulary has its own set of permissible Concept Classes, although the same Concept Class can be used by more than one Vocabulary. Depending on the Vocabulary, the Concept Class may categorize Concepts vertically (parallel) or horizontally (hierarchically). See the specification of each vocabulary for details.
8	Concept Class attributes should not be confused with Classification Concepts. These are separate Concepts that have a hierarchical relationship to Standard Concepts or each other, while Concept Classes are unique Vocabulary-specific attributes for each Concept.
9	For Concepts inherited from published terminologies, the source code is retained in the concept_code field and can be used to reference the source vocabulary.
10	Standard Concepts (designated as 'S' in the standard_concept field) may appear in CDM tables in all *_concept_id fields, whereas Classification Concepts ('C') should not appear in the CDM data, but participate in the construction of the CONCEPT_ANCESTOR table and can be used to identify Descendants that may appear in the data. See CONCEPT_ANCESTOR table. Non-standard Concepts can only appear in *_source_concept_id fields and are not used in CONCEPT_ANCESTOR table. Please refer to the Standardized Vocabularies specifications for details of the Standard Concept designation.
11	All logical data elements associated with the various CDM tables (usually in the _type_concept_id field) are called Type Concepts, including defining characteristics, qualifying attributes etc. They are also stored as Concepts in the CONCEPT table. Since they are generated by OMOP, their is no meaningful concept_code.
12	The lifespan of a Concept is recorded through its valid_start_date, valid_end_date and the invalid_reason fields. This allows Concepts to correctly reflect at which point in time were defined. Usually, Concepts get deprecated if their meaning was deemed ambiguous, a duplication of another Concept, or needed revision for scientific reason. For example, drug ingredients get updated when different salt or isomer variants enter the market. Usually, drugs taken off the market do not cause a deprecation by the terminology vendor. Since observational data are valid with respect to the time they are recorded, it is key for the Standardized Vocabularies to provide even obsolete codes and maintain their relationships to other current Concepts.

No.	Convention Description
13	Concepts without a known instantiated date are assigned valid_start_date of '1-Jan-1970'.
14	Concepts that are not invalid are assigned valid_end_date of '31-Dec-2099'.
15	Deprecated Concepts (with a valid_end_date before the release date of the Standardized Vocabularies) will have a value of 'D' (deprecated without successor) or 'U' (updated). The updated Concepts have a record in the CONCEPT_RELATIONSHIP table indicating their active replacement Concept.
16	Values for CONCEPT_IDs generated as part of Standardized Vocabularies will be reserved from 0 to 2,000,000,000. Above this range, CONCEPT_IDs are available for local use and are guaranteed not to clash with future releases of the Standardized Vocabularies.

4.2 VOCABULARY

The VOCABULARY table includes a list of the Vocabularies collected from various sources or created de novo by the OMOP community. This reference table is populated with a single record for each Vocabulary source and includes a descriptive name and other associated attributes for the Vocabulary.

Field	Required	Type	Description
vocabulary_id	Yes	varchar(20)	A unique identifier for each Vocabulary, such as ICD9CM, SNOMED, Visit.
vocabulary_name	Yes	varchar(255)	The name describing the vocabulary, for example "International Classification of Diseases, Ninth Revision, Clinical Modification, Volume 1 and 2 (NCHS)" etc.
vocabulary_reference	Yes	varchar(255)	External reference to documentation or available download of the about the vocabulary.
vocabulary_version	No	varchar(255)	Version of the Vocabulary as indicated in the source.
vocabulary_concept_id	Yes	integer	A foreign key that refers to a standard concept identifier in the CONCEPT table for the Vocabulary the VOCABULARY record belongs to.

4.2.1 Conventions

No.	Convention Description
1	There is one record for each Vocabulary. One Vocabulary source or vendor can issue several Vocabularies, each of them creating their own record in the VOCABULARY table. However, the choice of whether a Vocabulary contains Concepts of different Concept Classes, or when these different classes constitute separate Vocabularies cannot precisely be decided based on the definition of what constitutes a Vocabulary. For example, the ICD-9 Volume 1 and 2 codes (ICD9CM, containing predominantly conditions and some procedures and observations) and the ICD-9 Volume 3 codes (ICD9Proc, containing predominantly procedures) are realized as two different Vocabularies. On the other hand, SNOMED-CT codes of the class Condition and those of the class Procedure are part of one and the same Vocabulary. Please refer to the Standardized Vocabularies specifications for details of each Vocabulary.
2	The VOCABULARY_ID field contains an alphanumerical identifier, that can also be used as the abbreviation of the Vocabulary name.
3	The record with VOCABULARY_ID = 'None' is reserved to contain information regarding the current version of the Entire Standardized Vocabularies.
4	The VOCABULARY_NAME field contains the full official name of the Vocabulary, as well as the source or vendor in parenthesis.
5	Each Vocabulary has an entry in the CONCEPT table, which is recorded in the VOCABULARY_CONCEPT_ID field. This is for purposes of creating a closed Information Model, where all entities in the OMOP CDM are covered by a unique Concept.

4.3 DOMAIN

The DOMAIN table includes a list of OMOP-defined Domains the Concepts of the Standardized Vocabularies can belong to. A Domain defines the set of allowable Concepts for the standardized fields in the CDM tables. For example, the "Condition" Domain contains Concepts that describe a condition of a patient, and these Concepts can only be stored in the condition_concept_id field of the CONDITION_OCCURRENCE and CONDITION_ERA tables. This reference table is populated with a single record for each Domain and includes a descriptive name for the Domain.

Field	Required	Type	Description
domain_id	Yes	varchar(20)	A unique key for each domain.
domain_name	Yes	varchar(255)	The name describing the Domain, e.g. "Condition", "Procedure", "Measurement" etc.

Field	Required	Type	Description
domain_concept_id	Yes	integer	A foreign key that refers to an identifier in the CONCEPT table for the unique Domain Concept the Domain record belongs to.

4.3.1 Conventions

No.	Convention Description
1	There is one record for each Domain. The domains are defined by the tables and fields in the OMOP CDM that can contain Concepts describing all the various aspects of the healthcare experience of a patient.
2	The DOMAIN_ID field contains an alphanumerical identifier, that can also be used as the abbreviation of the Domain.
3	The DOMAIN_NAME field contains the unabbreviated names of the Domain.
4	Each Domain also has an entry in the Concept table, which is recorded in the DOMAIN_CONCEPT_ID field. This is for purposes of creating a closed Information Model, where all entities in the OMOP CDM are covered by unique Concepts.
5	Versions prior to v5.0.0 of the OMOP CDM did not support the notion of a Domain.

4.4 CONCEPT_CLASS

The CONCEPT_CLASS table is a reference table, which includes a list of the classifications used to differentiate Concepts within a given Vocabulary. This reference table is populated with a single record for each Concept Class:

Field	Required	Type	Description
concept_class_id	Yes	varchar(20)	A unique key for each class.
concept_class_name	Yes	varchar(255)	The name describing the Concept Class, e.g. "Clinical Finding", "Ingredient", etc.
concept_class_concept_id	Yes	integer	A foreign key that refers to an identifier in the CONCEPT table for the unique Concept Class the record belongs to.

4.4.1 Conventions

No.	Convention Description
1	There is one record for each Concept Class. Concept Classes are used to create additional structure to the Concepts within each Vocabulary. Some Concept Classes are unique to a Vocabulary (for example 'Clinical Finding' in SNOMED), but others can be used across different Vocabularies. The separation of Concepts through Concept Classes can be semantically horizontal (each Class subsumes Concepts of the same hierarchical level, akin to sub-Vocabularies within a Vocabulary) or vertical (each Class subsumes Concepts of a certain kind, going across hierarchical levels). For example, Concept Classes in SNOMED are vertical: The classes "Procedure" and "Clinical Finding" define very granular to very generic Concepts. On the other hand, 'Clinical Drug' and 'Ingredient' Concept Classes define horizontal layers or strata in the RxNorm vocabulary, which all belong to the same concept of a Drug.
2	The CONCEPT_CLASS_ID field contains an alphanumerical identifier, that can also be used as the abbreviation of the Concept Class.
3	The CONCEPT_CLASS_NAME field contains the unabbreviated names of the Concept Class.
4	Each Concept Class also has an entry in the Concept table, which is recorded in the concept_class_concept_id field. This is for purposes of creating a closed Information Model, where all entities in the OMOP CDM are covered by unique Concepts.

4.5 CONCEPT_RELATIONSHIP

The CONCEPT_RELATIONSHIP table contains records that define direct relationships between any two Concepts and the nature or type of the relationship. Each type of a relationship is defined in the RELATIONSHIP table.

Field	Required	Type	Description
concept_id_1	Yes	integer	A foreign key to a Concept in the CONCEPT table associated with the relationship. Relationships are directional, and this field represents the source concept designation.
concept_id_2	Yes	integer	A foreign key to a Concept in the CONCEPT table associated with the relationship. Relationships are directional, and this field represents the destination concept designation.
relationship_id	Yes	varchar(20)	A unique identifier to the type or nature of the Relationship as defined in the RELATIONSHIP table.

Field	Required	Type	Description
valid_start_date	Yes	date	The date when the instance of the Concept Relationship is first recorded.
valid_end_date	Yes	date	The date when the Concept Relationship became invalid because it was deleted or superseded (updated) by a new relationship. Default value is 31-Dec-2099.
invalid_reason	No	varchar(1)	Reason the relationship was invalidated. Possible values are 'D' (deleted), 'U' (replaced with an update) or NULL when valid_end_date has the default value.

4.5.1 Conventions

No.	Convention Description
1	Relationships can generally be classified as hierarchical (parent-child) or non-hierarchical (lateral).
2	All Relationships are directional, and each Concept Relationship is represented twice symmetrically within the CONCEPT_RELATIONSHIP table. For example, the two SNOMED concepts of 'Acute myocardial infarction of the anterior wall' and 'Acute myocardial infarction' have two Concept Relationships: 1- 'Acute myocardial infarction of the anterior wall' 'Is a' 'Acute myocardial infarction', and 2- 'Acute myocardial infarction' 'Subsumes' 'Acute myocardial infarction of the anterior wall'.
3	There is one record for each Concept Relationship connecting the same Concepts with the same RELATIONSHIP_ID.
4	Since all Concept Relationships exist with their mirror image (concept_id_1 and concept_id_2 swapped, and the RELATIONSHIP_ID replaced by the REVERSE_RELATIONSHIP_ID from the RELATIONSHIP table), it is not necessary to query for the existence of a relationship both in the concept_id_1 and concept_id_2 fields.
5	Concept Relationships define direct relationships between Concepts. Indirect relationships through 3rd Concepts are not captured in this table. However, the CONCEPT_ANCESTOR table does this for hierachical relationships over several "generations" of direct relationships.

4.6 RELATIONSHIP

The RELATIONSHIP table provides a reference list of all types of relationships that can be used to associate any two concepts in the CONCEPT_RELATIONSHP table.

Field	Required	Type	Description
relationship_id	Yes	varchar(20)	The type of relationship captured by the relationship record.
relationship_name	Yes	varchar(255)	The text that describes the relationship type.
is_hierarchical	Yes	varchar(1)	Defines whether a relationship defines concepts into classes or hierarchies. Values are 1 for hierarchical relationship or 0 if not.
defines_ancestry	Yes	varchar(1)	Defines whether a hierarchical relationship contributes to the concept_ancestor table. These are subsets of the hierarchical relationships. Valid values are 1 or 0.
reverse_relationship_id	Yes	varchar(20)	The identifier for the relationship used to define the reverse relationship between two concepts.
relationship_concept_id	Yes	integer	A foreign key that refers to an identifier in the CONCEPT table for the unique relationship concept.

4.6.1 Conventions

No.	Convention Description
1	There is one record for each Relationship.
2	Relationships are classified as hierarchical (parent-child) or non-hierarchical (lateral)
3	They are used to determine which concept relationship records should be included in the computation of the CONCEPT_ANCESTOR table.
4	The RELATIONSHIP_ID field contains an alphanumerical identifier, that can also be used as the abbreviation of the Relationship.
5	The RELATIONSHIP_NAME field contains the unabbreviated names of the Relationship.
6	Relationships all exist symmetrically, i.e. in both direction. The RELATIONSHIP_ID of the opposite Relationship is provided in the REVERSE_RELATIONSHIP_ID field.
7	Each Relationship also has an equivalent entry in the Concept table, which is recorded in the RELATIONSHIP_CONCEPT_ID field. This is for purposes of creating a closed Information Model, where all entities in the OMOP CDM are covered by unique Concepts.
8	Hierarchical Relationships are used to build a hierarchical tree out of the Concepts, which is recorded in the CONCEPT_ANCESTOR table. For example, 'has_ingredient' is a Relationship between Concept of the Concept Class 'Clinical Drug' and those of 'Ingredient', and all Ingredients can be classified as the 'parental' hierarchical Concepts for the drug products they are part of. All 'Is a' Relationships are hierarchical.

No.	Convention Description
9	Relationships, also hierarchical, can be between Concepts within the same Vocabulary or those adopted from different Vocabulary sources.

4.7 CONCEPT_SYNONYM

The CONCEPT_SYNONYM table is used to store alternate names and descriptions for Concepts.

Field	Required	Type	Description
concept_id	Yes	Integer	A foreign key to the Concept in the CONCEPT table.
concept_synonym_name	Yes	varchar(1000)	The alternative name for the Concept.
language_concept_id	Yes	integer	A foreign key to a Concept representing the language.

4.7.1 Conventions

No.	Convention Description
1	The concept_synonym_name field contains a valid Synonym of a concept, including the description in the concept_name itself. i.e., each Concept has at least one Synonym in the CONCEPT_SYNONYM table. As an example, for a SNOMED-CT Concept, if the fully specified name is stored as the concept_name of the CONCEPT table, then the Preferred Term and Synonyms associated with the Concept are stored in the CONCEPT_SYNONYM table.
2	Only Synonyms that are active and current are stored in the CONCEPT_SYNONYM table. Tracking synonym/description history and mapping of obsolete synonyms to current Concepts/Synonyms is out of scope for the Standard Vocabularies.
3	Currently, only English Synonyms are included.

4.8 CONCEPT_ANCESTOR

The CONCEPT_ANCESTOR table is designed to simplify observational analysis by providing the complete hierarchical relationships between Concepts. Only direct parent-child relationships between Concepts are stored in the CONCEPT_RELATIONSHIP table. To determine higher level ancestry connections, all individual direct relationships would have to be navigated at analysis time. The CONCEPT_ANCESTOR table includes records for all parent-child relationships, as well as grandparent-grandchild relationships and those of any other level of lineage. Using the CONCEPT_ANCESTOR table allows for querying for all descendants of a hierarchical concept. For example, drug ingredients and drug products are all descendants of a drug class ancestor.

This table is entirely derived from the CONCEPT, CONCEPT_RELATIONSHIP and RELATIONSHIP

tables.

Field	Required	Type	Description
ancestor_concept_id	Yes	integer	A foreign key to the concept in the concept table for the higher-level concept that forms the ancestor in the relationship.
descendant_concept_id	Yes	integer	A foreign key to the concept in the concept table for the lower-level concept that forms the descendant in the relationship.
min_levels_of_separation	Yes	integer	The minimum separation in number of levels of hierarchy between ancestor and descendant concepts. This is an attribute that is used to simplify hierarchic analysis.
max_levels_of_separation	Yes	integer	The maximum separation in number of levels of hierarchy between ancestor and descendant concepts. This is an attribute that is used to simplify hierarchic analysis.

4.8.1 Conventions

No.	Convention Description
1	Each concept is also recorded as an ancestor of itself.
2	Only valid and Standard Concepts participate in the CONCEPT_ANCESTOR table. It is not possible to find ancestors or descendants of deprecated or Source Concepts.
3	Usually, only Concepts of the same Domain are connected through records of the CONCEPT_ANCESTOR table, but there might be exceptions.

4.9 SOURCE_TO_CONCEPT_MAP

The source to concept map table is a legacy data structure within the OMOP Common Data Model, recommended for use in ETL processes to maintain local source codes which are not available as Concepts in the Standardized Vocabularies, and to establish mappings for each source code into a Standard Concept as target_concept_ids that can be used to populate the Common Data Model tables. The SOURCE_TO_CONCEPT_MAP table is no longer populated with content within the Standardized Vocabularies published to the OMOP community.

Field	Required	Type	Description
source_code	Yes	varchar(50)	The source code being translated into a Standard Concept.
source_concept_id	Yes	integer	A foreign key to the Source Concept that is being translated into a Standard Concept.

Field	Required	Type	Description
source_vocabulary_id	Yes	varchar(20)	A foreign key to the VOCABULARY table defining the vocabulary of the source code that is being translated to a Standard Concept.
source_code_description	No	varchar(255)	An optional description for the source code. This is included as a convenience to compare the description of the source code to the name of the concept.
target_concept_id	Yes	integer	A foreign key to the target Concept to which the source code is being mapped.
target_vocabulary_id	Yes	varchar(20)	A foreign key to the VOCABULARY table defining the vocabulary of the target Concept.
valid_start_date	Yes	date	The date when the mapping instance was first recorded.
valid_end_date	Yes	date	The date when the mapping instance became invalid because it was deleted or superseded (updated) by a new relationship. Default value is 31-Dec-2099.
invalid_reason	No	varchar(1)	Reason the mapping instance was invalidated. Possible values are D (deleted), U (replaced with an update) or NULL when valid_end_date has the default value.

4.9.1 Conventions

No.	Convention Description
1	This table is no longer used to distribute mapping information between source codes and Standard Concepts for the Standard Vocabularies. Instead, the CONCEPT_RELATIONSHIP table is used for this purpose, using the relationship_id='Maps to'.
2	However, this table can still be used for the translation of local source codes into Standard Concepts.
4	**Note:** This table should not be used to translate source codes to Source Concepts. The source code of a Source Concept is captured in its concept_code field. If the source codes used in a given database do not follow correct formatting the ETL will have to perform this translation. For example, if ICD-9-CM codes are recorded without a dot the ETL will have to perform a lookup function that allows identifying the correct ICD-9-CM Source Concept (with the dot in the concept_code field).

No.	Convention Description
5	The SOURCE_CONCEPT_ID, or the combination of the fields source_code and the SOURCE_VOCABULARY_ID uniquely identifies the source information. It is the equivalent to the CONCEPT_ID_1 field in the CONCEPT_RELATIONSHIP table.
6	If there is no SOURCE_CONCEPT_ID available because the source codes are local and not supported by the Standard Vocabulary, the content of the field is 0 (zero, not null) encoding an undefined concept. However, local Source Concepts are established (concept_id values above 2,000,000,000).
7	The SOURCE_CODE_DESCRIPTION contains an optional description of the source code.
8	The TARGET_CONCEPT_ID contains the Concept the source code is mapped to. It is equivalent to the concept_id_2 in the CONCEPT_RELATIONSHIP table
9	The TARGET_VOCABULARY_ID field contains the VOCABULARY_ID of the target concept. It is a duplication of the same information in the CONCEPT record of the Target Concept.
10	The fields VALID_START_DATE, VALID_END_DATE and INVALID_REASON are used to define the life cycle of the mapping information. Invalid mapping records should not be used for mapping information.

4.10 DRUG_STRENGTH

The DRUG_STRENGTH table contains structured content about the amount or concentration and associated units of a specific ingredient contained within a particular drug product. This table is supplemental information to support standardized analysis of drug utilization.

Field	Required	Type	Description
drug_concept_id	Yes	integer	A foreign key to the Concept in the CONCEPT table representing the identifier for Branded Drug or Clinical Drug Concept.
ingredient_concept_id	Yes	integer	A foreign key to the Concept in the CONCEPT table, representing the identifier for drug Ingredient Concept contained within the drug product.
amount_value	No	float	The numeric value associated with the amount of active ingredient contained within the product.
amount_unit_concept_id	No	integer	A foreign key to the Concept in the CONCEPT table representing the identifier for the Unit for the absolute amount of active ingredient.

Field	Required	Type	Description
numerator_value	No	float	The numeric value associated with the concentration of the active ingredient contained in the product
numerator_unit_concept_id	No	integer	A foreign key to the Concept in the CONCEPT table representing the identifier for the numerator Unit for the concentration of active ingredient.
denominator_value	No	float	The amount of total liquid (or other divisible product, such as ointment, gel, spray, etc.).
denominator_unit_concept_id	No	integer	A foreign key to the Concept in the CONCEPT table representing the identifier for the denominator Unit for the concentration of active ingredient.
box_size	No	integer	The number of units of Clinical of Branded Drug, or Quantified Clinical or Branded Drug contained in a box as dispensed to the patient
valid_start_date	Yes	date	The date when the Concept was first recorded. The default value is 1-Jan-1970.
valid_end_date	Yes	date	The date when the concept became invalid because it was deleted or superseded (updated) by a new Concept. The default value is 31-Dec-2099.
invalid_reason	No	varchar(1)	Reason the concept was invalidated. Possible values are 'D' (deleted), 'U' (replaced with an update) or NULL when valid_end_date has the default value.

4.10.1 Conventions

No.	Convention Description
1	The DRUG_STRENGTH table contains information for each active (non-deprecated) standard drug concept.
2	A drug which contains multiple active Ingredients will result in multiple DRUG_STRENGTH records, one for each active ingredient.
3	Ingredient strength information is provided either as absolute amount (usually for solid formulations) or as concentration (usually for liquid formulations).
4	If the absolute amount is provided (for example, 'Acetaminophen 5 MG Tablet') the AMOUNT_VALUE and AMOUNT_UNIT_CONCEPT_ID are used to define this content (in this case 5 and 'MG').

No.	Convention Description
5	If the concentration is provided (for example 'Acetaminophen 48 MG/ML Oral Solution') the NUMERATOR_VALUE in combination with the NUMERATOR_UNIT_CONCEPT_ID and DENOMINATOR_UNIT_CONCEPT_ID are used to define this content (in this case 48, 'MG' and 'ML').
6	In case of Quantified Clinical or Branded Drugs the DENOMINATOR_VALUE contains the total amount of the solution (not the amount of the ingredient). In all other drug concept classes the denominator amount is NULL because the concentration is always normalized to the unit of the denominator. So, a product containing 960 mg in 20 mL is provided as 48 mg/mL in the Clinical Drug and Clinical Drug Component, while as a Quantified Clinical Drug it is written as 960 mg/20 mL.
7	If the strength is provided in % (volume or mass-percent are not distinguished) it is stored in the NUMERATOR_VALUE/NUMERATOR_UNIT_CONCEPT_ID field combination, with both the DENOMINATOR_VALUE and DENOMINATOR_UNIT_CONCEPT_ID set to NULL. If it is a Quantified Drug the total amount of drug is provided in the DENOMINATOR_VALUE/DENOMINATOR_UNIT_CONCEPT_ID pair. E.g., the 30 G Isoconazole 2% Topical Cream is provided as 2% / in Clinical Drug and Clinical Drug Component, and as 2% /30 G.
8	Sometimes, one Ingredient is listed with different units within the same drug. This is very rare, and usually this happens if there are more than one Precise Ingredient. For example, 'Penicillin G, Benzathine 150000 UNT/ML / Penicillin G, Procaine 150000 MEQ/ML Injectable Suspension' contains Penicillin G in two different forms.
9	Sometimes, different ingredients in liquid drugs are listed with different units in the DENOMINATOR_UNIT_CONCEPT_ID. This is usually the case if the ingredients are liquids themselves (concentration provided as mL/mL) or solid substances (mg/mg). In these cases, the general assumptions is made that the density of the drug is that of water, and one can assume 1 g = 1 mL.
10	All Drug vocabularies containing Standard Concepts have entries in the DRUG_STRENGTH table.
11	There is now a Concept Class for supplier information whose relationships can be found in CONCEPT_RELATIONSHIP with a RELATIONSHIP_ID of 'Has supplier' and 'Supplier of'

5 Standardized Metadata

CDM_SOURCE METADATA

All metadata about the data should be derived from the data themselves.

5.1 CDM_SOURCE

The CDM_SOURCE table contains detail about the source database and the process used to transform the data into the OMOP Common Data Model.

Field	Required	Type	Description
cdm_source_name	Yes	varchar(255)	The full name of the source
cdm_source_abbreviation	No	varchar(25)	An abbreviation of the name
cdm_holder	No	varchar(255)	The name of the organization responsible for the development of the CDM instance
source_description	No	CLOB	A description of the source data origin and purpose for collection. The description may contain a summary of the period of time that is expected to be covered by this dataset.
source_documentation_reference	No	varchar(255)	URL or other external reference to location of source documentation
cdm_etl_reference	No	varchar(255)	URL or other external reference to location of ETL specification documentation and ETL source code
source_release_date	No	date	The date for which the source data are most current, such as the last day of data capture
cdm_release_date	No	date	The date when the CDM was instantiated
cdm_version	No	varchar(10)	The version of CDM used
vocabulary_version	No	varchar(20)	The version of the vocabulary used

5.1.1 Conventions

No.	Convention Description
1	If a source database is derived from multiple data feeds, the integration of those disparate sources is expected to be documented in the ETL specifications. The source information on each of the databases can be represented as separate records in the CDM_SOURCE table.
2	Currently, there is no mechanism to link individual records in the CDM tables to their source record in the CDM_SOURCE table.
3	The version of the vocabulary can be obtained from the vocabulary_name field in the VOCABULARY table for the record where vocabulary_id='None'.

5.2 METADATA

The METADATA table contains metadata information about a dataset that has been transformed to the OMOP Common Data Model.

Field	Required	Type	Description
metadata_concept_id	Yes	integer	A foreign key that refers to a Standard Metadata Concept identifier in the Standardized Vocabularies.
metadata_type_concept_id	Yes	integer	A foreign key that refers to a Standard Type Concept identifier in the Standardized Vocabularies.
name	Yes	varchar(250)	The name of the Concept stored in metadata_concept_id or a description of the data being stored.
value_as_string	No	nvarchar	The metadata value stored as a string.
value_as_concept_id	No	integer	A foreign key to a metadata value stored as a Concept ID.
metadata date	No	date	The date associated with the metadata
metadata_datetime	No	datetime	The date and time associated with the metadata

5.2.1 Conventions

No.	Convention Description
1	One record in the Metadata table is pre-populated in the DDL indicating the CDM version of the database.

6 Standardized Clinical Data Tables

PERSON
OBSERVATION_PERIOD
DEATH
VISIT_OCCURRENCE
VISIT_DETAIL
CONDITION_OCCURRENCE
DRUG_EXPOSURE
PROCEDURE_OCCURRENCE
DEVICE_EXPOSURE
MEASUREMENT
NOTE
NOTE_NLP
SURVEY_CONDUCT
OBSERVATION SPECIMEN
FACT_RELATIONSHIP

These tables contain the core information about the clinical events that occurred longitudinally during valid Observation Periods for each Person, as well as demographic information for the Person. Below provides an entity-relationship diagram highlighting the tables within the Standardized Clinical Data portion of the OMOP Common Data Model:

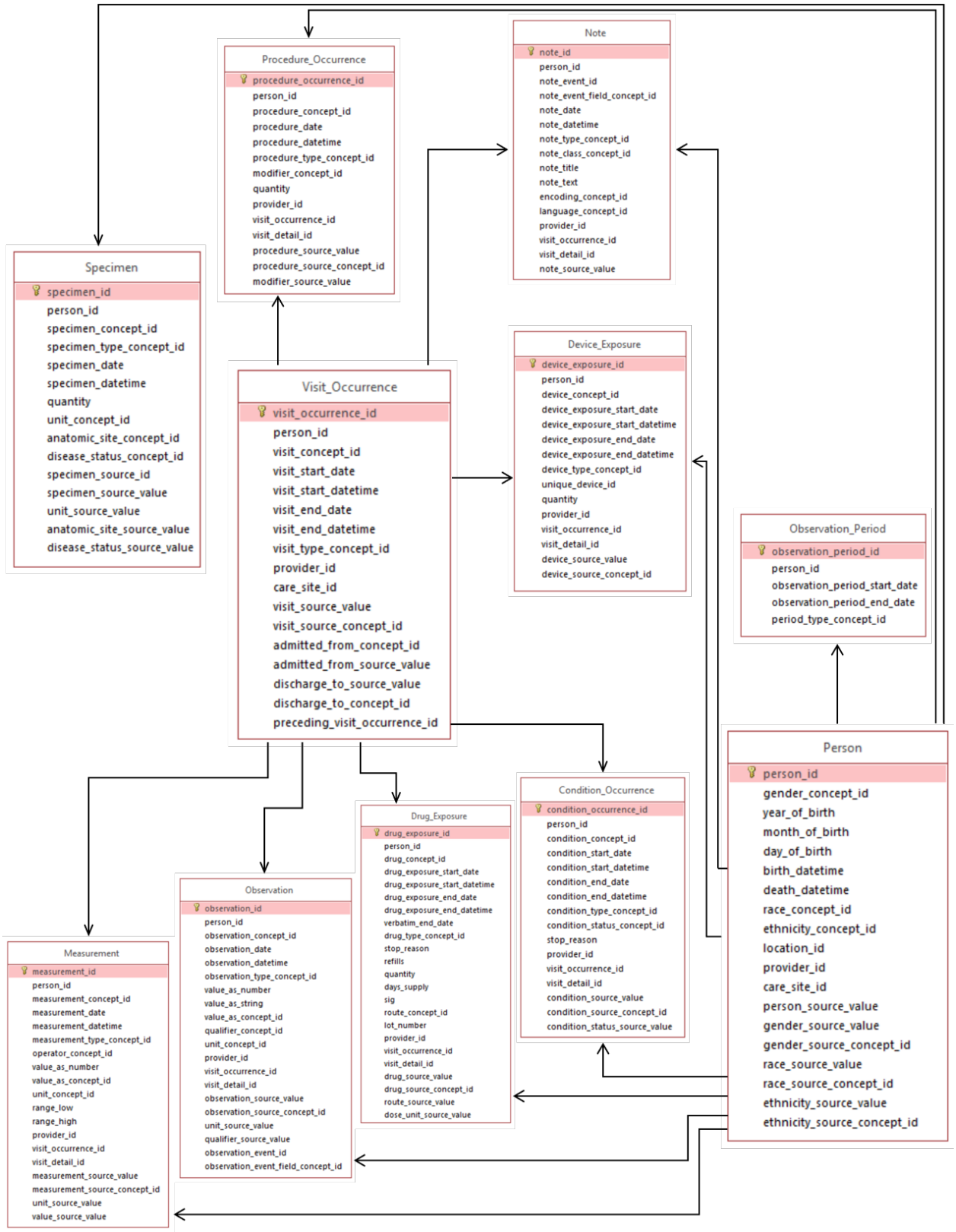
Procedure_Occurrence
procedure_occurrence_id
person_id
procedure_concept_id
procedure_date
procedure_datetime
procedure_type_concept_id
modifier_concept_id
quantity
provider_id
visit_occurrence_id
visit_detail_id
procedure_source_value
procedure_source_concept_id
modifier_source_value
Note
note_id
person_id
note_event_id
note_event_field_concept_id
note_date
note_datetime
note_type_concept_id
note_class_concept_id
note_title
note_text
encoding_concept_id
language_concept_id
provider_id
visit_occurrence_id
visit_detail_id
note_source_value
Specimen
specimen_id
person_id
specimen_concept_id
specimen_type_concept_id
specimen_date
specimen_datetime
quantity
unit_concept_id
anatomic_site_concept_id
disease_status_concept_id
specimen_source_id
specimen_source_value
unit_source_value
anatomic_site_source_value
disease_status_source_value
Visit_Occurrence
visit_occurrence_id
person_id
visit_concept_id
visit_start_date
visit_start_datetime
visit_end_date
visit_end_datetime
visit_type_concept_id
provider_id
care_site_id
visit_source_value
visit_source_concept_id
admitted_from_concept_id
admitted_from_source_value
discharge_to_source_value
discharge_to_concept_id
preceding_visit_occurrence_id
Device_Exposure
device_exposure_id
person_id
device_concept_id
device_exposure_start_date
device_exposure_start_datetime
device_exposure_end_date
device_exposure_end_datetime
device_type_concept_id
unique_device_id
quantity
provider_id
visit_occurrence_id
visit_detail_id
device_source_value
device_source_concept_id
Observation_Period
observation_period_id
person_id
observation_period_start_date
observation_period_end_date
period_type_concept_id
Person
person_id
gender_concept_id
year_of_birth
month_of_birth
day_of_birth
birth_datetime
death_datetime
race_concept_id
ethnicity_concept_id
location_id
provider_id
care_site_id
person_source_value
gender_source_value
gender_source_concept_id
race_source_value
race_source_concept_id
ethnicity_source_value
ethnicity_source_concept_id
Condition_Occurrence
condition_occurrence_id
person_id
condition_concept_id
condition_start_date
condition_start_datetime
condition_end_date
condition_end_datetime
condition_type_concept_id
condition_status_concept_id
stop_reason
provider_id
visit_occurrence_id
visit_detail_id
condition_source_value
condition_source_concept_id
condition_status_source_value
Drug_Exposure
drug_exposure_id
person_id
drug_concept_id
drug_exposure_start_date
drug_exposure_start_datetime
drug_exposure_end_date
drug_exposure_end_datetime
verbatim_end_date
drug_type_concept_id
stop_reason
refills
quantity
days_supply
sig
route_concept_id
lot_number
provider_id
visit_occurrence_id
visit_detail_id
drug_source_value
drug_source_concept_id
route_source_value
dose_unit_source_value
Observation
observation_id
person_id
observation_concept_id
observation_date
observation_datetime
observation_type_concept_id
value_as_number
value_as_string
value_as_concept_id
qualifier_concept_id
unit_concept_id
provider_id
visit_occurrence_id
visit_detail_id
observation_source_value
observation_source_concept_id
unit_source_value
qualifier_source_value
observation_event_id
observation_event_field_concept_id
Measurement
measurement_id
person_id
measurement_concept_id
measurement_date
measurement_datetime
measurement_type_concept_id
operator_concept_id
value_as_number
value_as_concept_id
unit_concept_id
range_low
range_high
provider_id
visit_occurrence_id
visit_detail_id
measurement_source_value
measurement_source_concept_id
unit_source_value
value_source_value

Figure 2:

6.1 PERSON

The Person Domain contains records that uniquely identify each patient in the source data who is time at-risk to have clinical observations recorded within the source systems.

Field	Required	Type	Description
person_id	Yes	integer	A unique identifier for each person.
gender_concept_id	Yes	integer	A foreign key that refers to an identifier in the CONCEPT table for the unique gender of the person.
year_of_birth	Yes	integer	The year of birth of the person. For data sources with date of birth, the year is extracted. For data sources where the year of birth is not available, the approximate year of birth is derived based on any age group categorization available.
month_of_birth	No	integer	The month of birth of the person. For data sources that provide the precise date of birth, the month is extracted and stored in this field.
day_of_birth	No	integer	The day of the month of birth of the person. For data sources that provide the precise date of birth, the day is extracted and stored in this field.
birth_datetime	No	datetime	The date and time of birth of the person.
death_datetime	No	datetime	The date and time of death of the person.
race_concept_id	Yes	integer	A foreign key that refers to an identifier in the CONCEPT table for the unique race of the person, belonging to the 'Race' vocabulary.
ethnicity_concept_id	Yes	integer	A foreign key that refers to the standard concept identifier in the Standardized Vocabularies for the ethnicity of the person, belonging to the 'Ethnicity' vocabulary.
location_id	No	integer	A foreign key to the place of residency for the person in the location table, where the detailed address information is stored.
provider_id	No	integer	A foreign key to the primary care provider the person is seeing in the provider table.
care_site_id	No	integer	A foreign key to the site of primary care in the care_site table, where the details of the care site are stored.
person_source_value	No	varchar(50)	An (encrypted) key derived from the person identifier in the source data. This is necessary when a use case requires a link back to the person data at the source dataset.
gender_source_value	No	varchar(50)	The source code for the gender of the person as it appears in the source data. The personâ€™s gender is mapped to a standard gender concept in the Standardized Vocabularies; the original value is stored here for reference.
gender_source_concept_id	Yes	Integer	A foreign key to the gender concept that refers to the code used in the source.

Field	Required	Type	Description
race_source_value	No	varchar(50)	The source code for the race of the person as it appears in the source data. The person race is mapped to a standard race concept in the Standardized Vocabularies and the original value is stored here for reference.
race_source_concept_id	Yes	Integer	A foreign key to the race concept that refers to the code used in the source.
ethnicity_source_value	No	varchar(50)	The source code for the ethnicity of the person as it appears in the source data. The person ethnicity is mapped to a standard ethnicity concept in the Standardized Vocabularies and the original code is, stored here for reference.
ethnicity_source_concept_id	Yes	Integer	A foreign key to the ethnicity concept that refers to the code used in the source.

6.1.1 Conventions

No.	Convention Description
1	All tables representing patient-related Domains have a foreign-key reference to the person_id field in the PERSON table.
2	Each person record has associated demographic attributes which are assumed to be constant for the patient throughout the course of their periods of observation. For example, the location or gender is expected to have a unique value per person, even though in life these data may change over time.
3	The GENDER_CONCEPT_ID should store what is believed to be the biological or sex assigned at birth. If the data set does have gender identification information, this should be stored in the OBSERVATION table (using the gender concepts 8532-Female or 8507-Male in OBSERVATION_CONCEPT_ID (THEMIS issue #32).
4	If we do not know the month or day of birth, we do not guess. A person can exist without a month or day of birth. If a person lacks a birth year that person should be dropped (THEMIS issue #30).
5	Living patients should not have a value in PERSON.DEATH_DATETIME, nor should they have any records relating to death either in the CONDITION_OCCURRENCE or OBSERVATION tables.
6	Only one death date per individual can be used. If a patient has clinical activity (e.g. prescriptions filled, labs performed, etc) more than 60+ days after death you may want to drop the death record as it may have been falsely reported. If multiple records of death exist on multiple days you may select the death that you deem most reliable (e.g. death at discharge) or select the latest death date.
7	If multiple death records occur, the date and the person have to be the same, but the cause can be different. Can be reported by different sources as well.
8	If PERSON.DEATH_DATETIME cannot be precisely determined from the data, the best approximation should be used.
9	The DEATH_DATETIME in the PERSON table should not be used as the way to find all deaths • `select * from PERSON where death_datetime is not null` should not be the practice • Rather, deaths should be found through the OBSERVATION table and the PERSON table is only used to determine which death date should be used in analysis.

No.	Convention Description
10	Valid Gender, Race and Ethnicity Concepts each belong to their own Domain.
11	Ethnicity in the OMOP CDM follows the OMB Standards for Data on Race and Ethnicity: Only distinctions between Hispanics and Non-Hispanics are made.
12	Additional information is stored through references to other tables, such as the home address (location_id) or the primary care provider.
13	The Provider refers to the primary care provider (General Practitioner). When the primary provider is unknown for a person then leave the PROVIDER_ID blank (THEMIS issue #36).
14	The Care Site refers to where the Provider typically provides the primary care. When care site for the primary provider is unknown then leave the CARE_SITE_ID blank.
15	It is not required that all subjects from the raw data be carried over to the CDM, in fact removing people that are not of high enough quality may help researchers using the CDM. Example scenarios to remove subjects include: a personâ€™s year of birth or age are unreasonable (e.g. born in year 0, 1800, 2999 or just lacking a year of birth), person lacks health benefits in claims database (i.e. thus you do not have a complete picture of their record), or raw data states that the person may not be of high research quality (e.g. CPRD will actually suggest which people not to use within research). Removal of a patient is not required and should be made in consideration of the raw data source. Reasons for removal of persons should be documented in the ETL documentation and METADATA table (insert row in METADATA where metadata.name='count of removed persons' and metada.value_as_string='xyz' where xyz is a number (e.g., 12). An ETL should not delete persons who contribute time however have no health care utilization (e.g. an individual enrolled in insurance but does not visit a doctor or pharmacy). This individual will contribute to analysis however as a healthy / non-care seeking individual (THEMIS issue #9).