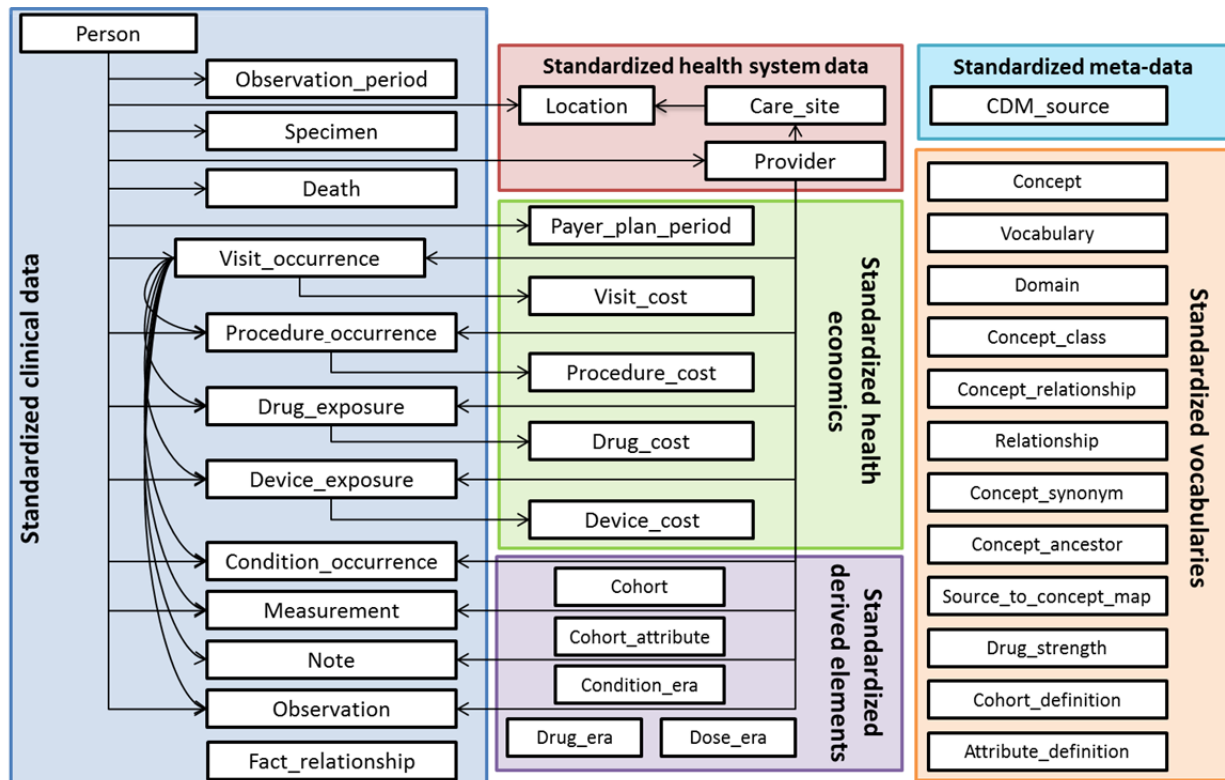


OMOP Common Data Model Specifications Version 5.0

October 14, 2014



License

© 2014 Observational Health Data Sciences and Informatics

This work is based on work by the Observational Medical Outcomes Partnership (OMOP) and used under license from the FNIH at <http://omop.fnih.org/publiclicense>.

All derivative work after the OMOP CDM v4 specification is dedicated to the public domain. Observational Health Data Sciences and Informatics (OHDSI) has waived all copyright and related or neighboring rights to the extent allowed by law.



<http://creativecommons.org/publicdomain/zero/1.0/>

Table of Contents

License	2
1 Background	5
1.1 The Role of the Common Data Model	5
1.2 Design Principles	6
1.3 Details of the Model	6
1.4 Data Model Conventions	13
2 Glossary of Terms	18
3 Standardized Vocabularies	22
3.1 CONCEPT	24
3.2 VOCABULARY	25
3.3 DOMAIN	26
3.4 CONCEPT_CLASS	26
3.5 CONCEPT_RELATIONSHIP	27
3.6 RELATIONSHIP	27
3.7 CONCEPT_SYNONYM	28
3.8 CONCEPT_ANCESTOR	28
3.9 SOURCE_TO_CONCEPT_MAP	29
3.10 DRUG_STRENGTH	30
3.11 COHORT_DEFINITION	31
3.12 ATTRIBUTE_DEFINITION	32
4 Standardized meta-data	33
4.1 CDM_SOURCE	33
5 Standardized Clinical Data Tables	35
5.1 PERSON	36
5.2 OBSERVATION_PERIOD	37
5.3 SPECIMEN	38
5.4 DEATH	39
5.5 VISIT_OCCURRENCE	40
5.6 PROCEDURE_OCCURRENCE	41
5.7 DRUG_EXPOSURE	43
5.8 DEVICE_EXPOSURE	45
5.9 CONDITION_OCCURRENCE	46
5.10 MEASUREMENT	47
5.11 NOTE	49

5.12	OBSERVATION	50
5.13	FACT_RELATIONSHIP	51
6	Standardized Health System Data Tables	53
6.1	LOCATION	54
6.2	CARE_SITE	54
6.3	PROVIDER	55
7	Standardized Health Economics Data Tables	57
7.1	PAYER_PLAN_PERIOD	58
7.2	VISIT_COST	58
7.3	PROCEDURE_COST	59
7.4	DRUG_COST	61
7.5	DEVICE_COST	62
8	Standardized Derived Elements	64
8.1	COHORT	65
8.2	COHORT_ATTRIBUTE	65
8.3	DRUG_ERA	66
8.4	DOSE_ERA	67
8.5	CONDITION_ERA	68

1 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:

- 1) Conduct methodological research to empirically evaluate the performance of various analytical methods on their ability to identify true associations and avoid false findings,
- 2) Develop tools and capabilities for transforming, characterizing, and analyzing disparate data sources across the health care delivery spectrum, and
- 3) 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 OMOP 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 OMOP Research Lab, a central computing resource developed to facilitate methodological research, has been transitioned to the Reagan-Udall Foundation for the FDA under the Innovation in Medical Evidence Development and Surveillance (IMEDS) Program, and has been re-branded as the IMEDS Lab. Learn more at: <http://imeds.reaganudall.org/>

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 <http://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.

1.1 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.

1.2 Design Principles

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

1. **Suitability for purpose.** The CDM aims at providing data organized in a way optimal for analysis, rather than for the purpose of operational needs of health care providers or payers.
2. **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.
3. **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.
4. **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.
5. **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.
6. **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.
7. **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.
8. **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.
9. **Backwards compatibility.** All changes from previous CDMs are clearly delineated. Older versions of the CDM can be easily created from this CDMv5, and no information is lost that was present previously.

The CDM is designed to include all observational health data elements 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.

1.3 Details of the Model

To represent the relevant domains, the CDM contains the following 39 tables:

Table name	Description
Standardized Vocabularies	
CONCEPT	The CONCEPT table contains records that uniquely identify each fundamental unit of meaning used to express clinical information. Concepts are derived from source vocabularies, which represent clinical information across different domains (e.g. conditions, drugs, procedures) through the use of source codes and associated descriptions. Some concepts are designated as standard concepts, meaning these concepts can be used within the OMOP Common Data Model and within standardized analytics. Each standard concept has a primary domain, which defines the location where the concept would be expected to be observed within OMOP Common Data Model.

Table name	Description
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.
DOMAIN	The DOMAIN table includes a list of the domains of data elements that are contained within the OMOP common data model. A domain defines the set of allowable concepts for each standardized field. This reference table is populated with a single record for each domain and includes a descriptive name for the Domain.
CONCEPT_CLASS	The CONCEPT_CLASS table 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 and includes a descriptive name for the concept class.
CONCEPT_RELATIONSHIP	The CONCEPT_RELATIONSHIP table contains records that define direct relationships between any two concepts and the nature of the relationship. The type of relationship is defined in the Relationship table, and is generally classified as hierarchical (parent-child) or non-hierarchical (lateral). 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'.
RELATIONSHIP	The RELATIONSHIP table provides a reference list of all allowable types of relationships that can be used to associate any two concepts in the concept relationship table. Relationships are classified as hierarchical (parent-child) or non-hierarchical, and are used to determine which concept relationship records should be included in the computation of the concept ancestor table.
CONCEPT_SYNONYM	The CONCEPT_SYNONYM table is used to store alternate names and descriptions for a concept. Each synonym is assigned its own unique identifier and contains the text of a description and the identifier of the concept that it represents.

Table name	Description
CONCEPT_ANCESTOR	The CONCEPT_ANCESTOR table contains records that define the inferred hierarchical relationships between all standard concepts. The concept ancestor table is fully derived from the concept, CONCEPT_RELATIONSHIP, and relationship tables, whereby all ancestor-descendant relationships can be inferred from traversing all parent-child relationships between standard concepts. The concept ancestor table includes records for all parent-child relationships, as well as grandparent-grandchild relationships and additional levels of lineage. The concept ancestor table enables efficient identification of multi-step hierarchical relationships, such as branded drugs that fall within a therapeutic class or specific diagnosis that are classified within a particular system organ class.
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 extract, transform, and load (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 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.
DRUG_STRENGTH	The DRUG_STRENGTH table contains structured content about the amount or concentration and associated units of a specific ingredient within a particular drug product. The drug strength table is a supplemental file to support standardized analysis of drug utilization using concepts from the RxNorm vocabulary. A clinical drug concept which contains multiple active ingredients will result in one drug strength record for each active ingredient.
COHORT_DEFINITION	The COHORT_DEFINITION table contains records to define each derived cohort through an associated description and syntax. Cohorts are derived elements of a set of subjects that satisfy a given set of inclusion criteria for a duration of time. The COHORT_DEFINITION table provides a standardized structure for maintaining the rules governing the inclusion of a subject into a cohort, and can store operational programming code to instantiate the cohort within a OMOP common data model.
ATTRIBUTE_DEFINITION	The ATTRIBUTE_DEFINITION table contains records to define each attribute through an associated description and syntax. Attributes are derived elements that can be selected or calculated for a subject within a cohort. The ATTRIBUTE_DEFINITION table provides a standardized structure for maintaining the rules governing the calculation of covariates for a subject in a cohort, and can store operational programming code to instantiate the attributes for a given cohort within the OMOP Common Data Model.
Standardized meta-data	
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. 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.
Standardized clinical data	

Table name	Description
PERSON	The PERSON table contains records that uniquely identify each patient in the source data who has time at-risk to have clinical events recorded within the source systems. A person must have at least one observation period to defined the time-at-risk but may or may not have any clinical events recorded in the other data domains. Each person record has associated demographic attributes which are assumed to be constant for the patient throughout the course of their periods of observation. All other patient-level data domains have a foreign-key reference to the person domain.
OBSERVATION_PERIOD	The OBSERVATION_PERIOD table contains records which uniquely define the spans of time for which a person is at-risk to have clinical events recorded within the source systems. One person may have one or more disjoint observation periods, during which times analyses may assume that clinical events would be captured if observed, and outside of which no clinical events may be recorded.
SPECIMEN	The SPECIMEN table contains the records identifying each biological sample from a person.
DEATH	The DEATH table contains the clinical event for how and when a person dies. A person can have up to one record if the source systems contain evidence that s/he is deceased. All persons who were alive during all observation periods should not contain any information in the death table.
VISIT_OCCURRENCE	The VISIT_OCCURRENCE table contains the spans of time a person continuously receives medical services from one or more providers at a facility in a given setting within the health care system. Visits are classified into 4 settings: outpatient care, inpatient confinement, emergency room, and long-term care. Persons may transition between these settings over the course of an episode of care. Inpatient visits are defined by the span of time between admission and discharge from a specific hospital facility. Outpatient visits are defined as span of time within a specific provider's office, which is expected to less than 1 day. Long-term care visits are defined as the span of time a person is treated within a specific long-term care facility.
PROCEDURE_OCCURRENCE	The PROCEDURE_OCCURRENCE table contains records of activities or processes ordered by and/or carried out by a healthcare provider on the patient to have a diagnostic and/or therapeutic purpose.
DRUG_EXPOSURE	The DRUG_EXPOSURE table captures records about the inferred utilization of a biochemical substance with a physiological therapeutic effect when ingested or otherwise introduced into the body. Drugs include prescription and over-the-counter medicines, vaccines, and large-molecule biologic therapies. Drug exposure is inferred from clinical events associated with orders, prescriptions written, pharmacy dispensings, procedural administrations, and other patient-reported information.

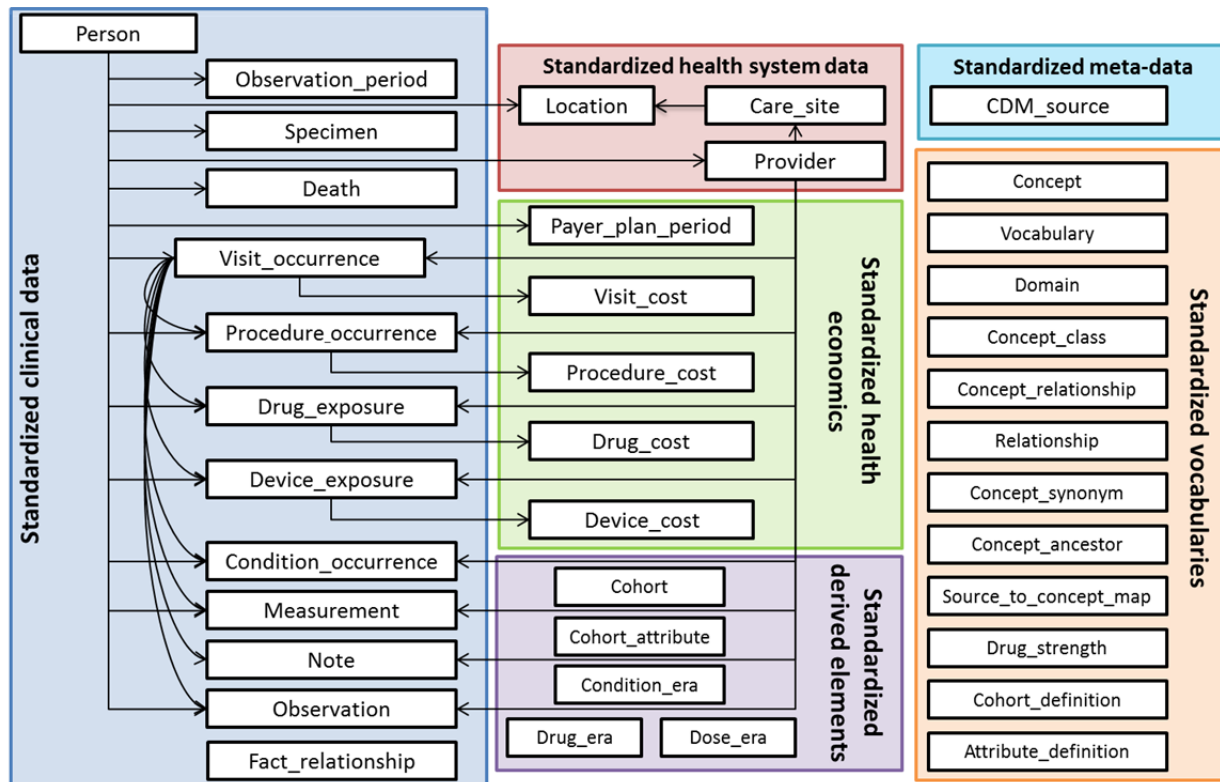
Table name	Description
DEVICE_EXPOSURE	The DEVICE_EXPOSURE table captures records about a person's inferred exposure to a foreign physical object or instrument that is used for diagnostic or therapeutic purposes through a mechanism beyond chemical action. Devices include implantable objects (e.g. pacemakers, stents, artificial joints), durable medical equipment and supplies (e.g. bandages, crutches, syringes), and other instruments used in medical procedures (e.g. sutures, defibrillators).
CONDITION_OCCURRENCE	The CONDITION_OCCURRENCE table captures records of a disease or a medical condition based on evaluation by a provider or reported by a patient.
MEASUREMENT	A measurement is the capture of a structured value (numerical or categorical) obtained through systematic examination of a person or sample. The MEASUREMENT table captures measurement orders and measurement results. The measurement domain can contain laboratory results, vital signs, or quantitative findings from pathology reports.
NOTE	The NOTE table captures unstructured information that was recorded by a provider or a patient in free text notes on a given date.
OBSERVATION	The OBSERVATION table captures any clinical facts about a patient obtained in the context of examination, questioning or a procedure. The observation domain supports capture of data not represented by other domains, including unstructured measurements, medical history and family history.
FACT_RELATIONSHIP	The FACT_RELATIONSHIP table contains records to detail the relationships between facts within one domain or across two domains, and the nature of the relationship. Examples of types of fact relationships include: person relationships (mother-child linkage), care site relationships (representing the hierarchical organization structure of facilities within health systems), drug exposures provided due to associated indicated condition, devices used during the course of an associated procedure, and measurements derived from an associated specimen. All relationships are directional, and each relationship is represented twice symmetrically within the fact relationship table. For example, two persons (PERSON_ID = 1 is the mother of PERSON_ID = 2) have two fact relationships: 1- 'PERSON_ID 1' 'parent of' 'PERSON_ID 2', and 2- 'PERSON_ID 2' 'child of' 'PERSON_ID 1'.
Standardized health system data	
LOCATION	The LOCATION table represents a generic way to capture physical location or address information. Locations are used to define the addresses for Persons and Care Sites.

Table name	Description
CARE_SITE	The CARE_SITE table contains a list of uniquely identified physical or organizational units where healthcare delivery is practiced (offices, wards, hospitals, clinics, etc.).
PROVIDER	The PROVIDER table contains a list of uniquely identified health care providers. These are typically physicians and nurses.
Standardized health economics	
PAYER_PLAN_PERIOD	The PAYER_PLAN_PERIOD table captures records that detail the period of time that a person is continuously enrolled under a specific health plan benefit structure from a given payer. Each Person receiving health care and covered by a health benefits is subject to a Plan defined by the Payer for the Person or her family. For a given benefit policy, there may be one or more Plans that are active for certain periods of time (e.g. before and after the deductible is reached), determining the cost of health services provided.
VISIT_COST	The VISIT_COST table captures the costs of health visit of a patient which are not itemized to specific procedures, drugs, or devices used within the encounter.
PROCEDURE_COST	The PROCEDURE_COST table captures the cost of a Procedure performed on a Person. The information about the cost is only derived from the amounts paid for the Procedure.
DRUG_COST	The DRUG_COST table captures records indicating the cost of a Drug Exposure. The information about the cost is defined by the amount of money paid by the person and payer for the drug, as well as the charged cost of the drug.
DEVICE_COST	The DEVICE_COST table captures the cost of a medical device or supply used on a Person. The information about the cost is only derived from the amounts paid for the device.
Standardized derived elements	
COHORT	The COHORT table contains records derived as a set of subjects that satisfy a given set of inclusion criteria for a duration of time. The definition of the cohort is contained within the COHORT_DEFINITION table. Example cohorts can include patients diagnosed with a specific condition, patients exposed to a particular drug, or providers who have performed a specific procedure.

Table name	Description
COHORT_ATTRIBUTE	The COHORT_ATTRIBUTE table contains attributes associated with each subject within a cohort, as defined by a given set of inclusion criteria for a duration of time. The definition of the cohort attribute is contained within the ATTRIBUTE_DEFINITION table. Example cohort attributes can be age, BMI or comorbidity score.
DRUG_ERA	A Drug Era is defined as a span of time when the Person is assumed to be exposed to a particular active ingredient. A Drug Era is not the same as a Drug Exposure: Exposures are individual records corresponding to the source when drug was delivered to the Person, while successive periods of Drug Exposures are combined under certain rules to produce continuous Drug Eras.
DOSE_ERA	A Dose Era is defined as a span of time when the Person is assumed to be exposed to a constant dose of a specific active ingredient.
CONDITION_ERA	A Condition Era is defined as a span of time when the Person is assumed to have a given condition.

The CDM defines table structures for each of the data in a Person-centric. The tables have foreign keys into the Person table and a date. This allows for a longitudinal view on all the healthcare-relevant events. The exceptions from this rule are the standardized health system data tables. Providers carrying out health care are linked to many of the events as well. Both are linked to healthcare organizations (hospitals, independent physician associations), care sites (doctor's offices, hospital departments etc.) and physical locations (addresses).

OMOP CDM v5 Conceptual Model:



1.4 Data Model Conventions

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

1.4.1 GENERAL CONVENTIONS OF DATA TABLES

The CDM is platform-independent. Data types are defined generically using ANSI SQL data types (VARCHAR, INTEGER, FLOAT, DATE, TIME, 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 time format. Standard queries against CDM may vary for local instantiations and date/time 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.

1.4.2 GENERAL CONVENTIONS OF FIELDS

Variable names across all tables follow one convention:

- <entity>_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

- Ex: condition_source_value = '787.02' was the ICD-9 code captured as a diagnosis from the administrative claim
- <entity>_ID:
 - Unique identifiers for key entities, which can serve as foreign keys to establish relationships across entities
 - Ex: person_id uniquely identifies each individual. visit_occurrence_id uniquely identifies a PERSON encounter at a point of care.
- <entity>_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
 - Ex: condition_concept_id = 31967 contains reference value for SNOMED concept of 'Nausea'
- <entity>_SOURCE_CONCEPT_ID
 - Foreign key into the Standardized Vocabularies representing the concept and terminology used in the source data, when applicable
 - Ex: condition_source_concept_id = 35708202 denotes the concept of 'Nausea' in the MedDRA terminology; the analogous condition_concept_id might be 31967, since SNOMED-CT is the Standardized Vocabularies for most clinical diagnoses and findings.
- <entity>_TYPE_CONCEPT_ID:
 - Delineates the origin of the source information, standardized within the Standardized Vocabularies
 - Ex: drug_type_concept_id can allow analysts to discriminate between 'Pharmacy dispensing' and 'Prescription written'

1.4.3 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 unmappable concept.

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

1.4.4 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 contains the codes from public code systems such as ICD-9-CM, NDC, CPT-4 etc. or local controlled vocabularies (such as F for female and M for male) copied from the source data. Source Values are stored in the _source_value field 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. 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, but not for controlled vocabularies. 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.

1.4.5 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).

1.4.6 TIME SPAN OF AVAILABLE DATA

Data tables for clinical data contain a date stamp (ending in _date, _start_date or _end_date), 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 happened prior the first OBSERVATION_PERIOD, it will be captured as a record in the OBSERVATION table of 'Medical history' (concept_id = 43054928), with the observation_date 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.

1.4.7 CONTENT OF EACH TABLE

For the tables of the main domains of the CDM it is imperative that used concepts 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: V25.5 "Encounter for insertion of implantable subdermal contraceptive" defines a procedure and is therefore stored in the PROCEDURE_OCCURRENCE table.

1.4.8 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, are stored as unstructured text, and are generally not to be used by any standardized analytics. 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 the preferred fields to use 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:

1. Map all source values directly to standard concept_ids. Store these mappings in the SOURCE_TO_CONCEPT_MAP table.
 - If the source code is not mappable to a vocabulary term, the source_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:

1. Map all your source values to the corresponding concept_ids in the source vocabulary. 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 mappable to a vocabulary term, the source_concept_id field is set to 0
2. 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_id. Each standard concept_id belongs to only 1 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. Ex: HCPCS code for infusion of a drug, will map to a concept in the procedure domain of the infusion and a different concept in the drug domain for the product infused. It is also possible for one source_concept_id to map to multiple standard concept_ids within the same domain. Ex: ICD-9 for 'viral hepatitis with hepatic coma' maps to SNOMED 'viral hepatitis' and a different 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.
3. Write the data record into table(s) corresponding to the domain of the standard concept_id(s).

- If the source value is mapped to source_concept_id, but the source_concept_id is not mapped to a standard concept_id, then the domain for the data record is determined by the CONCEPT_RELATIONSHIP from the source_concept_id to a domain concept and 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. (Ex: if the unmappable source_value came from a 'diagnosis' table, then in the absence of other information, you may choose to record that data 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).

There is no constraint on allowed concept_ids within the source_concept_id fields.

1.4.81.4.9 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.

2 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.
Ambulatory Payment Classification	APC	The Ambulatory Payment Classifications is used as a method of paying for outpatient services for the Medicare program, which is analogous to the DRGs for inpatient services.
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.
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).
Current Procedural Terminology, 4th edition	CPT-4	A terminology that is maintained by the American Medical Association (AMA). It is used by hospitals for Medicare hospital outpatient and by physicians for outpatient services.
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.

Term	Abbr.	Description
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.
Diagnosis-Related Group	DRG	The Diagnosis-related Groups is used as a method of paying for inpatient services for the Medicare program, which is analogous to the APCs for outpatient services.
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).
Generic Product Identifier	GPI	A proprietary unique identifier for a drug used by the commercial Medi-Span® formulary database.
Healthcare Common Procedure Coding System	HCPCS	HCPCS Level I codes are managed by the AMA (licensing fees apply). The HCPCS Level II codes are managed by CMS (Centers for Medicare & Medicaid Services). The Level II codes include: alphanumeric HCPCS procedure and modifier codes, their long and short descriptions, and applicable Medicare administrative, coverage, and pricing data. These codes are used for Medicare outpatient services.
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.

Term	Abbr.	Description
Health Level Seven	HL7	HL7 is a global not-for-profit, ANSI-accredited standards developing organization dedicated to providing a comprehensive framework and related standards for the exchange, integration, sharing, and retrieval of electronic health information that supports clinical practice and the management, delivery and evaluation of health services. HL7 specifications primarily draw upon codes and vocabularies from a variety of sources.
International Classification of Disease, 9th Revision, Clinical Modifications	ICD-9-CM Or ICD-9	The official system of assigning codes to diagnoses and procedures associated with hospital utilization in the United States.
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.
Logical Observation Identifiers Names and Codes	LOINC	Universal code names and identifiers to medical terminology related to the Electronic Health Record and assists in the electronic exchange and gathering of clinical results (such as laboratory tests, clinical observations, outcomes management and research).
Medical Dictionary for Regulatory Activities	MedDRA	MedDRA is a clinically-validated international medical terminology used by regulatory authorities and the regulated biopharmaceutical industry. The terminology is used through the entire regulatory process, from pre-marketing to post-marketing, and for data entry, retrieval, evaluation, and presentation.
National Drug Codes	NDC	Unique identifiers assigned to individual drugs. NDCs are used primarily as an inventory code and for prescriptions.
National Drug File - Reference Terminology	NDF-RT	A nonproprietary drug reference terminology that includes drug knowledge and classifies drugs, most notably by mechanism of action and physiologic effect.
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.

Term	Abbr.	Description
RxNorm		<p>A standardized nomenclature for clinical drugs and drug delivery devices is produced by the National Library of Medicine. In RxNorm, the name of a clinical drug combines its ingredients, strengths, and/or form.</p> <p>RxNorm provides normalized names for clinical drugs and links its names to many of the drug vocabularies commonly used in pharmacy management and drug interaction software, including those of First DataBank, Micromedix, MediSpan, Gold Standard Alchemy, and Multum. By providing links between these vocabularies, RxNorm can mediate messages between systems not using the same software and vocabulary.</p>
Systematized Nomenclature of Medicine - Clinical Terms	SNOMED-CT®	<p>SNOMED-CT is one of a suite of designated standards for use in U.S. Federal Government systems for the electronic exchange of clinical health information, and is also a required standard in interoperability specifications of the U.S. Healthcare Information Technology Standards Panel. SNOMED-CT is also being implemented internationally as a standard within other IHTSDO Member countries.</p>
Terminology		<p>Technical or special terms used in a business or special subject area.</p>
Vocabulary		<p>A computerized list (as of items of data or words) used for reference (as for information retrieval or word processing).</p>

3 Standardized Vocabularies

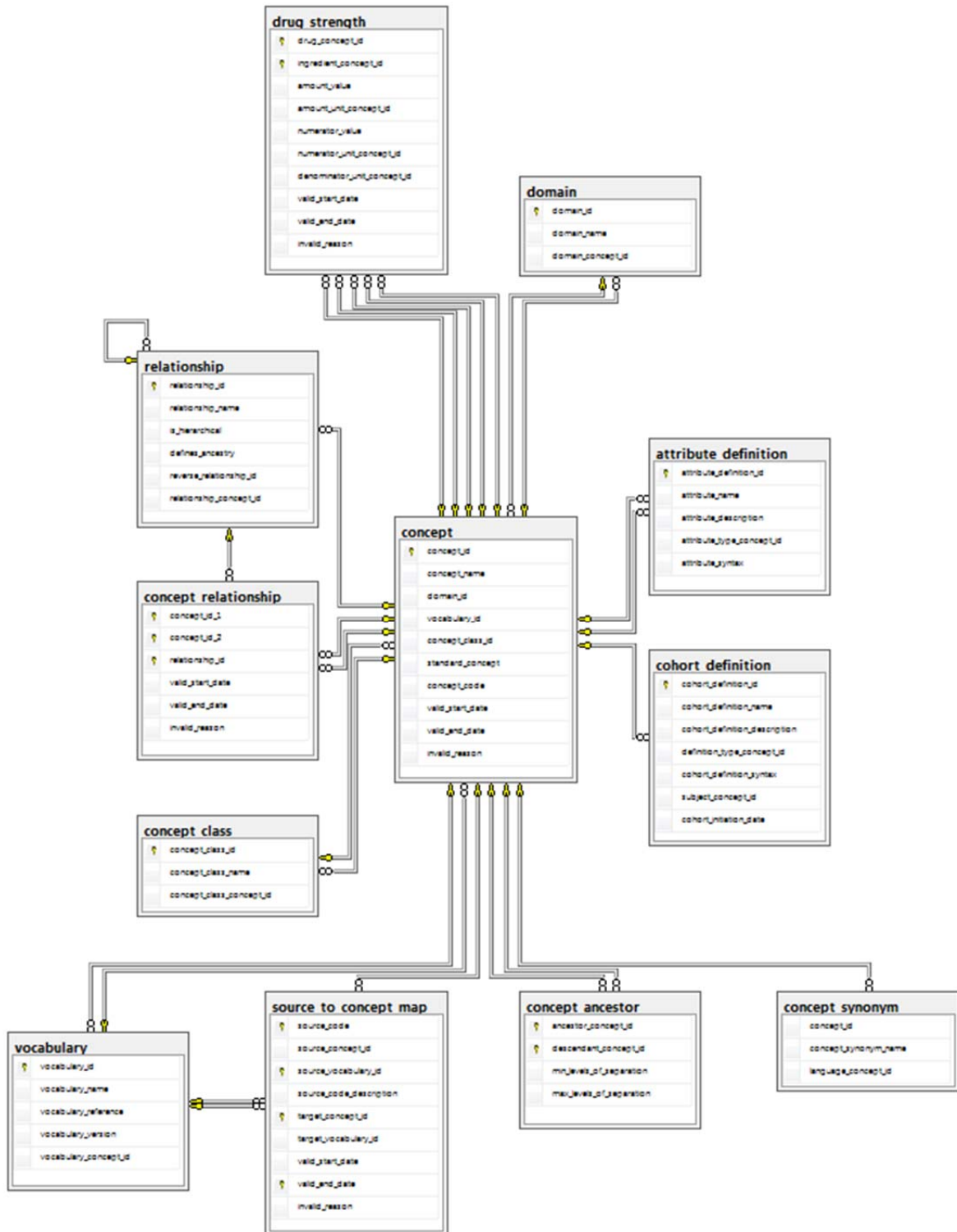
These tables contain detailed information about the Concepts used in all of the CDM fact tables. The content of the Standardized Vocabularies tables is not generated anew by each CDM implementation. Instead, it is maintained centrally as a service to the community.

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 OMOP 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:



Standardized Vocabularies entity-relationship diagram

3.1 CONCEPT

The Standardized Vocabularies contains records, or concepts, that uniquely identify each fundamental unit of meaning used to express clinical information. Concepts are derived from vocabularies, which represent clinical information across different domains (e.g. conditions, drugs, procedures) through the use of codes and associated descriptions. Some concepts are designated as 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 has a primary domain, which defines the location where the concept would be expected to occur within the OMOP Common Data Model.

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.

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)	The domain a 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 category or class of the concept along both the hierarchical tree as well as different domains within a vocabulary. 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 class concept, or a non-standard concept. The allowable values for standard_concept are: 'S' for Standard Concept, 'C' for a Class Concept, otherwise the content is null for non-standard concepts, including source concepts. Standard Concepts (S) may appear in CDM tables in CONCEPT_ID fields, whereas Class concepts (C) should not appear in the CDM data, but participate in the CONCEPT_ANCESTOR and can be used to identify descendants that may appear in the data. Non-standard concepts can only appear in SOURCE_CONCEPT_ID fields and are not used in CONCEPT_ANCESTOR.
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.
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.

Field	Required	Type	Description
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.

3.1.1 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:

- All Concepts are maintained centrally by OMOP. Additional concepts can be added, as needed, upon request.
- 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.
- The concept_id of a Concept is persistent, i.e. stays the same for the same Concepts between releases of the Standardized Vocabularies.
- 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.
- Each concept is assigned to one domain.
- For Concepts inherited from published terminologies, the source code is retained as part of the Concept reference data and used to reference the source vocabulary.
- All logical data elements associated with the various CDM tables, usually called Types, including defining characteristics, qualifying attributes etc. are also stored as Concepts. Since they are generated by OMOP, their source_code is omitted.
- The lifespan of concepts is recorded through their valid_start_date, valid_end_date and the invalid_reason. This allows concepts to correctly reflect at which point in time were in active clinical use. For example, drugs that are taken off the market might be dropped by the terminology vendor. However, 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 Concepts and Classifications.
- Concepts without a known instantiated date are assigned VALID_START_DATE of '1970-01-01'
- Concepts that are not invalid are assigned VALID_END_DATE of '2099-12-31'
- Concepts generated as part of Standardized Vocabularies will be reserved as values from 0 to 2,000,000,000.

3.2 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.

Vocabulary_id = 0 is reserved to contain information regarding the current version of the vocabulary.

Field	Required	Type	Description
vocabulary_id	Yes	varchar(20)	A unique identifier for each vocabulary.

Field	Required	Type	Description
vocabulary_name	Yes	varchar(255)	The name describing the vocabulary, for example "SNOMED-CT", "ICD-9", "Visit", etc.
vocabulary_reference	Yes	varchar(255)	External reference to documentation or available download of the about the vocabulary
vocabulary_version	Yes	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 Standardized Vocabularies for the vocabulary the Vocabulary record belongs to.

3.3 DOMAIN

The DOMAIN table includes a list of the domains of data elements that are contained within the OMOP common data model. A domain defines the set of allowable concepts for each standardized field. 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.
domain_concept_id	Yes	integer	A foreign key that refers to a standard concept identifier in the Standardized Vocabularies for the Domain the Domain record belongs to.

3.4 CONCEPT_CLASS

The CONCEPT_CLASS table 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 and includes a descriptive name for the 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 a standard concept identifier in the Standardized Vocabularies for the Class the record belongs to.

3.5 CONCEPT_RELATIONSHIP

The concept relationship table contains records that define direct relationships between any two concepts and the nature of the relationship. The type of relationship is defined in the Relationship table, and is generally classified as hierarchical (parent-child) or non-hierarchical (lateral). 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'.

Field	Required	Type	Description
concept_id_1	Yes	integer	A foreign key to the 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 the 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)	The type of relationship as defined in the relationship table.
valid_start_date	Yes	date	The date when the instance of the relationship is first recorded.
valid_end_date	Yes	date	The date when the 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.

3.6 RELATIONSHIP

The relationship table provides a reference list of all allowable types of relationships that can be used to associate any two concepts in the concept relationship table. Relationships are classified as hierarchical (parent-child) or non-hierarchical, and are used to determine which concept relationship records should be included in the computation of the concept ancestor table.

- Hierarchical Relationships are used to define a hierarchical tree between Concepts. For example, "has_ingredient" is a Relationship between Clinical Drugs and Ingredients, and all Ingredients can be classified as the "parental" hierarchical Concepts for the Drugs they are included in. All "is a" Relationships are hierarchical. Hierarchical Relationships can be between Concepts that are adopted from the same Vocabulary source or between Concepts adopted from different Vocabulary sources. Only certain Hierarchical Relationships are used to build the Concept Ancestor relationships.
- Non-hierarchical Relationships are all remaining non-inclusive relationships, for example between Clinical Drugs and Branded Drugs. These Relationships are not utilized to create Ancestor relationships.

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.

Field	Required	Type	Description
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 a standard concept identifier in the Standardized Vocabularies for the relationship concept.

3.7 CONCEPT_SYNONYM

The concept synonym table is used to store alternate names and descriptions for a concept. Each synonym is assigned its own unique identifier and contains the text of a description and the identifier of the concept that it represents.

Each Concept may be linked to zero or more Synonyms in the CONCEPT_SYNONYM table. As an example, for a SNOMED-CT Concept, if the fully specified name is stored as the Concept name in the CONCEPT table, then the Preferred Term and Synonyms associated with the Concept are stored in the CONCEPT_SYNONYM table.

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 OMOP vocabularies.

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.

3.8 CONCEPT_ANCESTOR

The Concept Ancestor table is a convenience table designed to simplify observational analysis by consolidating the hierarchical relationship between various Concepts. Parent-child relationships between Concepts are stored in the Concept Relationship table. However, it is stored in a form that is hard to navigate due to the interlocking nature of the relationships and the multiplicity of parent-child relationships for many Concepts. The concept_ancestor table contains records that define the inferred hierarchical relationships between all standard concepts. The concept ancestor table is fully derived from the concept, CONCEPT_RELATIONSHIP, and relationship tables, whereby all ancestor-descendant relationships can be inferred from traversing all parent-child relationships between standard concepts. The concept ancestor table includes records for all parent-child relationships, as well as grandparent-grandchild relationships and additional levels of lineage. A concept is also an ancestor of itself. The concept ancestor table enables efficient identification of multi-step hierarchical relationships, such as branded drugs that fall within a therapeutic class or specific diagnosis that are classified within a particular organ system class.

The Ancestor-Descendant relationship captures hierarchical relationships between Ancestor and any Descendant Concepts, along with indicators for the shortest and longest navigation path (Maximum and Minimum Levels of Separation) between them.

Only valid (non-invalidated) Standard Concepts of the same domain participate in the Concept Ancestor table.

The Ancestor relationship is primarily targeted at observational analysis that would involve:

- Rollup of lower level Concepts into higher-level aggregation Concepts.
- Collection of all lower level Concepts in the hierarchy that follow from a high level Concept.

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.

3.9 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 to which the source code is being mapped.
source_vocabulary_id	Yes	varchar(20)	A foreign key to the vocabulary table defining the vocabulary of the source code that is being mapped to the Standardized Vocabularies.
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.

Field	Required	Type	Description
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.

3.10 DRUG_STRENGTH

The drug strength table contains structured content about the amount or concentration and associated units of a specific ingredient within a particular drug product. The drug strength table is a supplemental file to support standardized analysis of drug utilization using concepts from the RxNorm vocabulary. A clinical drug concept which contains multiple active ingredients will result in one drug strength record for each active ingredient.

Field	Required	Type	Description
drug_concept_id	Yes	integer	A foreign key to the concept in the concept table, representing the identifier for an RxNorm 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 an RxNorm 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, in situations when the ingredient is contained within a solid formulation.
amount_unit_concept_id	No	integer	The unit corresponding to the numeric value of the amount of the active ingredient contained within the product.
numerator_value	No	float	The numeric value associated with the concentration of the active ingredient contained within the product, in situations when the ingredient is contained within a non-solid formulation

Field	Required	Type	Description
numerator_unit_concept_id	No	integer	The numerator unit corresponding to the numeric value of the concentration of the active ingredient contained within the product.
denominator_unit_concept_id	No	integer	The denominator unit corresponding to the numeric value of the concentration of the active ingredient contained within the product.
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.

3.11 COHORT_DEFINITION

The cohort definition table contains records to define each derived cohort through an associated description and syntax. Cohorts are derived elements of a set of subjects that satisfy a given set of inclusion criteria for a duration of time. The cohort definition table provides a standardized structure for maintaining the rules governing the inclusion of a subject into a cohort, and can store operational programming code to instantiate the cohort within the OMOP Common Data Model.

Field	Required	Type	Description
cohort_definition_id	Yes	integer	A unique identifier for each cohort.
cohort_definition_name	Yes	varchar(255)	A short description of the cohort.
cohort_definition_description	No	CLOB	A complete description of the cohort definition
definition_type_concept_id	Yes	integer	Type to define what kind of cohort definition the record represents and how the syntax may be executed
cohort_definition_syntax	No	CLOB	Syntax to operationalize the cohort definition
subject_concept_id	Yes	integer	A foreign key to the concept to which defines the domain of subjects that are members of the cohort (e.g., person, provider, visit).
cohort_instantiation_date	No	Date	A date to indicate when the cohort was instantiated in the COHORT table

3.12 ATTRIBUTE_DEFINITION

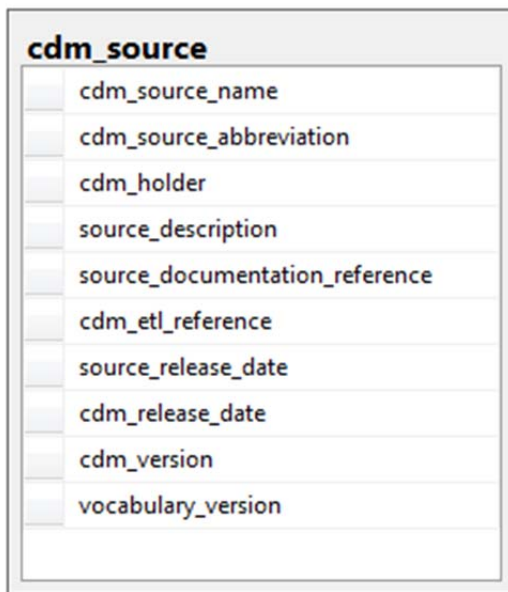
The attribute definition table contains records to define each attribute through an associated description and syntax. Attributes are derived elements that can be selected or calculated for a subject within a cohort. The attribute definition table provides a standardized structure for maintaining the rules governing the calculation of covariates for a subject in a cohort, and can store operational programming code to instantiate the attributes for a given cohort within the OMOP Common Data Model.

Field	Required	Type	Description
attribute_definition_id	Yes	integer	A unique identifier for each attribute.
attribute_name	Yes	varchar(255)	A short description of the attribute.
attribute_description	No	CLOB	A complete description of the attribute definition
attribute_type_concept_id	Yes	integer	Type to define what kind of attribute definition the record represents and how the syntax may be executed
attribute_syntax	No	CLOB	Syntax to operationalize the attribute definition

4 Standardized meta-data

All metadata about the data should be derived from the data themselves. However, the following contains a few key pieces of information that are convenient especially for software applications utilizing the CDM data.

Below provides an entity-relationship diagram highlighting the tables within the Standardized Meta-data portion of the OMOP Common Data Model:



4.1 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. 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.

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.

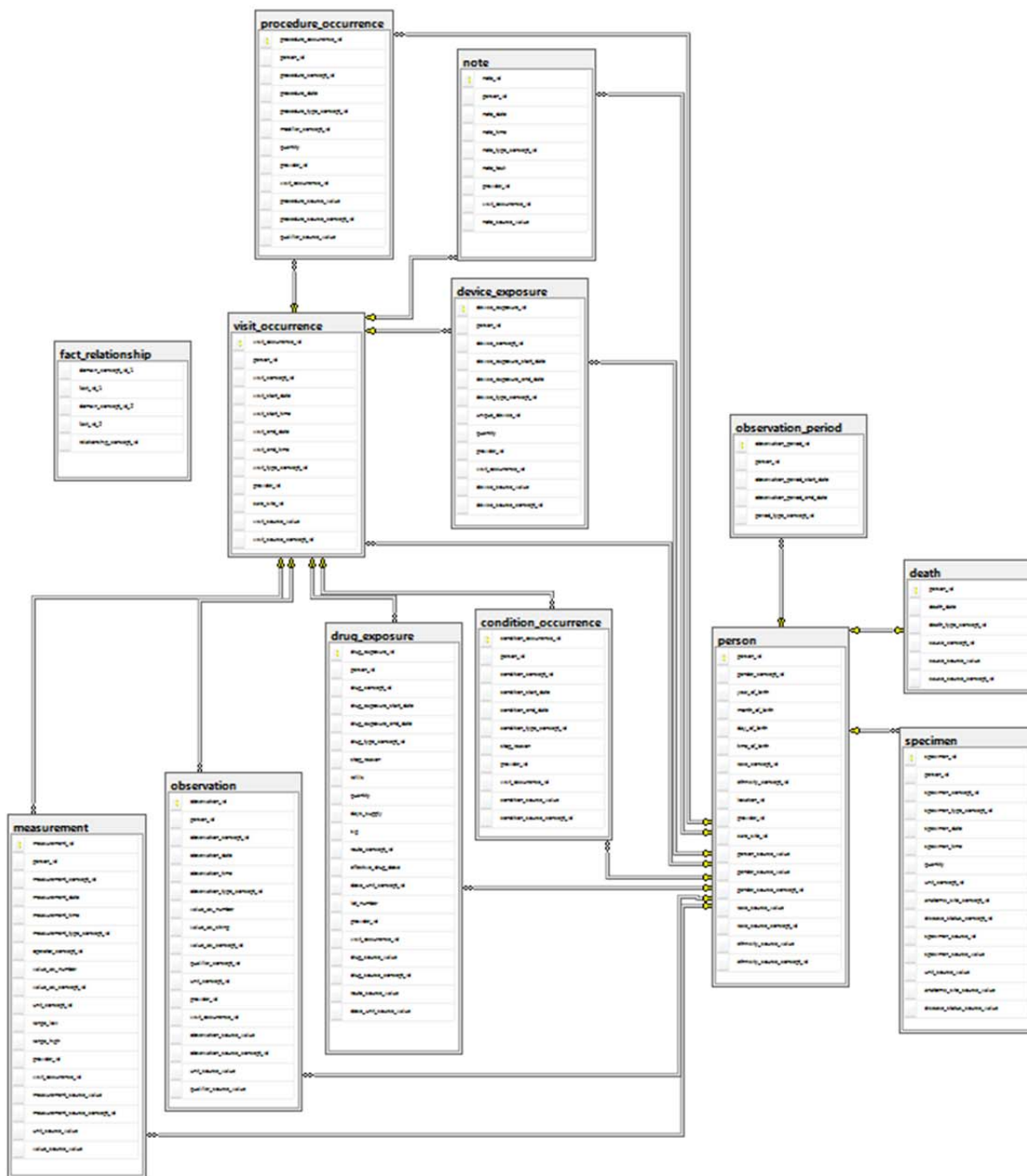
OMOP Common Data Model Specification – Version 5

Field	Required	Type	Description
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 is 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 Standardized Clinical Data Tables

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:



5.1 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. Each person record has associated demographic attributes which are assumed to be constant for the patient throughout the course of their periods of observation. All other patient-related data domains have a foreign-key reference to the person domain.

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 a standard concept identifier in the Standardized Vocabularies for the 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.
time_of_birth	No	Time	The time of birth at the birth day
race_concept_id	Yes	integer	A foreign key that refers to a standard concept identifier in the Standardized Vocabularies for the race of the person.
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.
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.

Field	Required	Type	Description
gender_source_value	No	varchar(50)	The source code for the gender of the person as it appears in the source data. The person'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	No	Integer	A foreign key to the gender concept that refers to the code used in the source.
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	No	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	No	Integer	A foreign key to the ethnicity concept that refers to the code used in the source.

5.1.1 CONVENTIONS

- Valid Gender, Race and Ethnicity Concepts belong to the "Demographic" domain.
- Person source data attributes are race, gender, and ethnicity.
- Ethnicity in the OMOP CDM follows the OMB Standards for Data on Race and Ethnicity: Only distinctions between Hispanics and Non-Hispanics are made.
- Additional information is stored through references to other tables about the home address (location_id) and the primary care provider.
- The provider refers to the primary care provider (General Practitioner).
- The care site refers to where the provider typically provides the primary care.
- All persons are required to have a valid gender and year of birth.
- The person table requires only one value for each attribute. While it is possible for a person to change genders, locations, and providers over time, it is the responsibility of the data holder to select the one value to use in the CDM.

5.2 OBSERVATION_PERIOD

The observation period domain contains records which uniquely define the spans of time for which a person is at-risk to have clinical events recorded within the source systems. One person may have one or more disjoint observation periods, during which times analyses may assume that clinical events would be captured if observed, and outside of which no clinical events may be recorded.

Field	Required	Type	Description
-------	----------	------	-------------

Field	Required	Type	Description
observation_period_id	Yes	integer	A unique identifier for each observation period.
person_id	Yes	integer	A foreign key identifier to the person for whom the observation period is defined. The demographic details of that person are stored in the person table.
observation_period_start_date	Yes	date	The start date of the observation period for which data are available from the data source.
observation_period_end_date	Yes	date	The end date of the observation period for which data are available from the data source.
period_type_concept_id	Yes	Integer	A foreign key identifier to the predefined concept in the Standardized Vocabularies reflecting the source of the observation period information

5.2.1 CONVENTIONS

- Each Person can have more than one valid OBSERVATION_PERIOD record, but no two observation periods can overlap in time for a given person.
- During an Observation Period, any clinical event that happens to the patient is expected to be recorded. Conversely, the absence of data indicates that no clinical events occurred to the patient.
- No clinical data are valid outside an active Observation Period. Clinical data that refer to a time outside (diagnoses of previous conditions such as "Old MI" or medical history) of an active Observation Period are recorded as Observations. The date of the Observation is the first day of the first Observation Period of a patient.
- For claims data, observation periods are inferred from the enrollment periods to a health benefit plan.

5.3 SPECIMEN

The specimen domain contains the records identifying each biological sample from a person.

Field	Required	Type	Description
specimen_id	Yes	integer	A unique identifier for each specimen.
person_id	Yes	integer	A foreign key identifier to the person for whom the specimen is recorded.
specimen_concept_id	Yes	integer	A foreign key referring to a standard concept identifier in the Standardized Vocabularies for the specimen.
specimen_type_concept_id	Yes	integer	A foreign key referring to the predefined concept identifier in the Standardized Vocabularies reflecting the system of record from which the specimen was represented in the source data.

Field	Required	Type	Description
specimen_date	Yes	date	The date the specimen was obtained from the person.
specimen_time	No	time	The time on the date when the specimen was obtained from the person.
quantity	No	float	The amount of specimen collection from the person during the sampling procedure
unit_concept_id	No	integer	A foreign key to a standard concept identifier for the unit associated with the numeric quantity of the specimen collection.
anatomic_site_concept_id	No	integer	A foreign key to a standard concept identifier for the anatomic location of specimen collection.
disease_status_concept_id	No	integer	A foreign key to a standard concept identifier for the disease status of specimen collection.
specimen_source_id	No	varchar(50)	The specimen identifier as it appears in the source data.
specimen_source_value	No	varchar(50)	The specimen value as it appears in the source data. This value is mapped to a standard concept in the Standardized Vocabularies and the original code is, stored here for reference.
unit_source_value	No	varchar(50)	The information about the unit as detailed in the source.
anatomic_site_source_value	No	varchar(50)	The information about the anatomic site as detailed in the source.
disease_status_source_value	No	varchar(50)	The information about the disease status as detailed in the source.

5.3.1 CONVENTIONS

- Anatomic site is coded at the most specific level of granularity possible, such that higher level classifications can be derived using the Standardized Vocabularies

5.4 DEATH

The death domain contains the clinical event for how and when a person dies. A person can have up to one record if the source systems contain evidence that s/he is deceased, such as:

- Condition Code in the Header or Detail information of claims
- Status of enrollment into a health plan
- Explicit record in EHR data

Living patients should not contain any information in the death table.

Field	Required	Type	Description
person_id	Yes	integer	A foreign key identifier to the deceased person. The demographic details of that person are stored in the person table.
death_date	Yes	date	The date the person was deceased. If the precise date including day or month is not known or not allowed, December is used as the default month, and the last day of the month the default day.
death_type_concept_id	Yes	integer	A foreign key referring to the predefined concept identifier in the Standardized Vocabularies reflecting how the death was represented in the source data.
cause_concept_id	No	integer	A foreign key referring to a standard concept identifier in the Standardized Vocabularies for conditions.
cause_source_value	No	varchar(50)	The source code for the cause of death as it appears in the source data. This code is mapped to a standard concept in the Standardized Vocabularies and the original code is, stored here for reference.
cause_source_concept_id	No	Integer	A foreign key to the concept that refers to the code used in the source. Note, this variable name is abbreviated to ensure it will be allowable across database platforms.

5.4.1 CONVENTIONS

- Each Person may have more than one record of death in the source data. It is the task of the ETL to pick the most plausible or most accurate records to be aggregated and stored as a single record in the Death table.
- If the Death Date cannot be precisely determined from the data, the best approximation should be used.

5.5 VISIT_OCCURRENCE

The visit domain contains the spans of time a person continuously receives medical services from one or more providers at a care site in a given setting within the health care system. Visits are classified into 4 settings: outpatient care, inpatient confinement, emergency room, and long-term care. Persons may transition between these settings over the course of an episode of care. If applicable, relationships between visits within an episode of care may be represented in the FACT_RELATIONSHIP table.

Visits are recorded in various data sources in different forms with varying levels of standardization. For example:

- Medical Claims include Inpatient Admissions, Outpatient Services, and Emergency Room visits.
- Electronic Health Records may capture Person visits as part of the activities recorded.

Field	Required	Type	Description
-------	----------	------	-------------

Field	Required	Type	Description
visit_occurrence_id	Yes	integer	A unique identifier for each person's visit or encounter at a healthcare provider.
person_id	Yes	integer	A foreign key identifier to the person for whom the visit is recorded. The demographic details of that person are stored in the person table.
visit_concept_id	Yes	integer	A foreign key that refers to a visit concept identifier in the Standardized Vocabularies.
visit_start_date	Yes	date	The start date of the visit.
visit_start_time	No	time	The time the visit started.
visit_end_date	Yes	date	The end date of the visit. If this is a one-day visit the end date should match the start date.
visit_end_time	No	time	The time the visit ended.
visit_type_concept_id	Yes	Integer	A foreign key to the predefined concept identifier in the Standardized Vocabularies reflecting the type of source data from which the visit record is derived.
provider_id	No	integer	A foreign key to the provider in the provider table who was associated with the visit.
care_site_id	No	integer	A foreign key to the care site in the care site table that was visited.
visit_source_value	No	Varchar(50)	The source code for the visit as it appears in the source data.
visit_source_concept_id	No	Integer	A foreign key to a concept that refers to the code used in the source.

5.5.1 CONVENTIONS

- A Visit Occurrence is recorded for each visit to a healthcare facility.
- Valid Visit Concepts belong to the "Visit" domain.
- Standard Visit Concepts are defined as Inpatient Visit, Outpatient Visit, Emergency Room Visit and Long Term Care Visit. Source concepts from place of service vocabularies are mapped into these standard visit concepts in the Standardized Vocabularies.
- Each Visit is standardized by assigning a corresponding Concept Identifier based on the type of facility visited and the type of services rendered.
- At any one day, there could be more than one visit.
- One visit may involve multiple providers, in which case the ETL must specify how a single provider id is selected or leave the provider_id field null.
- One visit may involve multiple care sites, in which case the ETL must specify how a single care_site id is selected or leave the care_site_id field null.

5.6 PROCEDURE_OCCURRENCE

The procedure domain contains records of activities or processes ordered by and/or carried out by a healthcare provider on the patient to have a diagnostic and/or therapeutic purpose.

Procedures are present in various data sources in different forms with varying levels of standardization. For example:

- Medical Claims include CPT-4, ICD-9-CM (Procedures), and HCPCS procedure codes that are submitted as part of a claim for health services rendered, including procedures performed.
- Electronic Health Records that capture CPT-4, ICD-9-CM (Procedures), HCPCS or OPCS-4 procedures as orders.

Field	Required	Type	Description
procedure_occurrence_id	Yes	integer	A system-generated unique identifier for each procedure occurrence.
person_id	Yes	integer	A foreign key identifier to the person who is subjected to the procedure. The demographic details of that person are stored in the person table.
procedure_concept_id	Yes	integer	A foreign key that refers to a standard procedure concept identifier in the Standardized Vocabularies.
procedure_date	Yes	date	The date on which the procedure was performed.
procedure_type_concept_id	Yes	integer	A foreign key to the predefined concept identifier in the Standardized Vocabularies reflecting the type of source data from which the procedure record is derived.
modifier_concept_id	No	integer	A foreign key to a standard concept identifier for a modifier to the procedure (e.g. bilateral)
quantity	No	integer	The quantity of procedures ordered or administered.
provider_id	No	integer	A foreign key to the provider in the provider table who was responsible for carrying out the procedure.
visit_occurrence_id	No	integer	A foreign key to the visit in the visit table during which the procedure was carried out.
procedure_source_value	No	varchar(50)	The source code for the procedure as it appears in the source data. This code is mapped to a standard procedure concept in the Standardized Vocabularies and the original code is stored here for reference. Procedure source codes are typically ICD-9-Proc, CPT-4, HCPCS or OPCS-4 codes.
procedure_source_concept_id	No	integer	A foreign key to a procedure concept that refers to the code used in the source.
qualifier_source_value	No	varchar(50)	The source code for the qualifier as it appears in the source data.

5.6.1 CONVENTIONS

- Valid Procedure Concepts belong to the "Procedure" domain. Procedure Concepts are based on a variety of vocabularies: SNOMED-CT, ICD-9-Proc, CPT-4, HCPCS and OPCS-4.
- Procedures are expected to be carried out within one day.
- Procedures could involve the application of a drug, in which case the procedural component is recorded in the procedure table and simultaneously the administered drug in the drug exposure table when both the procedural component and drug are identifiable.
- If the quantity value is omitted, a single procedure is assumed.
- The Procedure Type defines from where the Procedure Occurrence is drawn or inferred; for administrative claims records, the type indicates whether a Procedure was primary or secondary and their relative positioning within a claim.

- The Visit during which the procedure was performed is recorded through a reference to the VISIT_OCCURRENCE table. This information is not always available.
- The Provider carrying out the procedure is recorded through a reference to the PROVIDER table. This information is not always available.

5.7 DRUG_EXPOSURE

The drug exposure domain captures records about the inferred utilization of a biochemical substance with a physiological effect when ingested or otherwise introduced into the body. Drugs include prescription and over-the-counter medicines, vaccines, and large-molecule biologic therapies. Drug exposure is inferred from clinical events associated with orders, prescriptions written, pharmacy dispensings, procedural administrations, and other patient-reported information.

Drug Exposure records are recorded from a variety of source information:

- The “Prescription” section of an EHR captures prescriptions written by physicians or from electronic ordering systems
- The "Medication list" section of an EHR for both non-prescription products and medications prescribed by other providers
- Prescriptions filled at dispensing providers such as pharmacies, and then captured in reimbursement claim systems
- Drugs administered as part of a Procedure, such as chemotherapy or vaccines

Only drugs with active pharmaceutical ingredients are recorded. Radiological devices ingested or applied locally do not count as drugs.

Field	Required	Type	Description
drug_exposure_id	Yes	Integer	A system-generated unique identifier for each drug utilization event.
person_id	Yes	Integer	A foreign key identifier to the person who is subjected to the drug. The demographic details of that person are stored in the person table.
drug_concept_id	Yes	Integer	A foreign key that refers to a standard concept identifier in the Standardized Vocabularies for the drug concept.
drug_exposure_start_date	Yes	date	The start date for the current instance of drug utilization. Valid entries include a start date of a prescription, the date a prescription was filled, or the date on which a drug administration procedure was recorded.
drug_exposure_end_date	No	date	The end date for the current instance of drug utilization. It is not available from all sources.
drug_type_concept_id	Yes	integer	A foreign key to the predefined concept identifier in the Standardized Vocabularies reflecting the type of drug exposure recorded. It indicates how the drug exposure was represented in the source data: as medication history, filled prescriptions, etc.
stop_reason	No	varchar(20)	The reason the medication was stopped, where available. Reasons include regimen completed, changed, removed, etc.
refills	No	integer	The number of refills after the initial prescription. The initial prescription is not counted, values start with 0.

Field	Required	Type	Description
quantity	No	float	The quantity of drug as recorded in the original prescription or dispensing record.
days_supply	No	integer	The number of days of supply of the medication as recorded in the original prescription or dispensing record.
sig	No	CLOB	The directions ("signetur") on the drug prescription as recorded in the original prescription (and printed on the container) or dispensing record.
route_concept_id	No	integer	A foreign key to a predefined concept in the Standardized Vocabularies reflecting the route of administration.
effective_drug_dose	No	float	Numerical value of drug dose for this drug_exposure record.
dose_unit_concept_id	No	integer	A foreign key to a predefined concept in the Standardized Vocabularies reflecting the unit the effective_drug_dose value is expressed.
lot_number	No	varchar(50)	An identifier to determine where the product originated
provider_id	No	integer	A foreign key to the provider in the provider table who initiated (prescribed) the drug exposure.
visit_occurrence_id	No	integer	A foreign key to the visit in the visit table during which the drug exposure initiated.
drug_source_value	No	varchar(50)	The source code for the drug as it appears in the source data. This code is mapped to a standard drug concept in the Standardized Vocabularies and the original code is, stored here for reference.
drug_source_concept_id	No	Integer	A foreign key to a drug concept that refers to the code used in the source.
route_source_value	No	varchar(50)	The information about the route of administration as detailed in the source.
dose_unit_source_value	No	varchar(50)	The information about the dose unit as detailed in the source.

5.7.1 CONVENTIONS

- Valid Drug Concepts belong to the "Drug" domain. Most Concepts in the Drug domain are based on RxNorm, but some may come from other sources. Concepts are members of the Clinical Drug or Pack, Branded Drug or Pack, Drug Component or Ingredient classes.
- Source drug identifiers, including NDC codes, Generic Product Identifiers, etc. are mapped to standard drug Concepts in the Standardized Vocabularies (e.g., based on RxNorm). When the Drug Source Value of the code cannot be translated into standard Drug Concept IDs, a Drug exposure entry is stored with only the corresponding source_concept_id and drug_source_value and a drug_concept_id of 0.
- The Drug Concept with the highest content of information is preferred during the mapping process: Concept Classes Branded Drug or Pack, followed by Clinical Drug, followed by Drug Component, and only if no other information is available the Ingredient. If only the drug class is known, no drug record should be written.
- A Drug Type is assigned to each Drug Exposure to track from what source the data were drawn or inferred.
- The Effective Drug Dose and the Dose Unit Concepts are provided in cases when the dose is explicitly provided, as it is typically for pediatric and chemotherapeutic treatments, and can only

refer to a single active ingredient. Combination products which have doses for each ingredient need to be recorded as separate records.

- If possible, the visit in which the drug was prescribed or delivered is recorded through a reference to the visit table.

5.8 DEVICE_EXPOSURE

The device exposure domain captures records about a person's inferred exposure to a foreign physical object or instrument that which is used for diagnostic or therapeutic purposes through a mechanism beyond chemical action. Devices include implantable objects (e.g. pacemakers, stents, artificial joints), durable medical equipment and supplies (e.g. bandages, crutches, syringes), and other instruments used in medical procedures (e.g. sutures, defibrillators).

Field	Required	Type	Description
device_exposure_id	Yes	integer	A system-generated unique identifier for each device exposure.
person_id	Yes	integer	A foreign key identifier to the person who is subjected to the procedure. The demographic details of that person are stored in the person table.
device_concept_id	Yes	integer	Only the DI portion of the UDI would be captured as a Concept in the Standardized Vocabularies.
device_exposure_start_date	Yes	date	The date the device or supply was applied or used.
device_exposure_end_date	No	date	The date the device or supply was removed from use.
device_type_concept_id	Yes	integer	Provenance for the data, e.g. procedure device, from registry, etc.
unique_device_id	No	varchar(50)	The entire UDI or equivalent.
quantity	No	integer	The number of individual devices used for the exposure
provider_id	No	integer	A foreign key to the provider in the provider table who was responsible for using the device.
visit_occurrence_id	No	integer	A foreign key to the visit in the visit table during which the device was used.
device_source_value	No	varchar(50)	The source code for the device as it appears in the source data. This code is mapped to a standard device concept in the Standardized Vocabularies and the original code is stored here for reference.
device_source_concept_id	No	integer	A foreign key to a device concept that refers to the code used in the source.

5.8.1 CONVENTIONS

- Valid Device Concepts belong to the "Device" domain.
- The distinction between devices or supplies and procedures are sometimes blurry, but the former are physical objects while the latter are actions, often to apply a device or supply.
- For medical devices that are regulated by the FDA, a Unique Device Identification (UDI) is required if available in the data source, and is recorded in the unique_device_id field.
- The DI portion of that UDI is used to define concepts in the CONCEPT table. However, devices are also defined based on other source vocabularies, like HCPCS.
- The Visit during which the device was first used is recorded through a reference to the VISIT_OCCURRENCE table. This information is not always available.

- The Provider exposing the patient to the device is recorded through a reference to the PROVIDER table. This information is not always available.

5.9 CONDITION_OCCURRENCE

The condition occurrence domain captures records of clinical observations of a person suggestive of the existence of disease or a medical condition based on diagnoses, signs and/or symptoms observed by a provider or reported by a patient.

Conditions are recorded in different sources and levels of standardization. For example:

- Medical claims data include ICD-9-CM diagnosis codes that are submitted as part of a claim for health services and procedures.
- EHRs may capture Person conditions in the form of diagnosis codes and symptoms as ICD-9-CM codes, but may not have a way to capture out-of-system conditions.

Field	Required	Type	Description
condition_occurrence_id	Yes	integer	A unique identifier for each condition occurrence event.
person_id	Yes	integer	A foreign key identifier to the person who is experiencing the condition. The demographic details of that person are stored in the person table.
condition_concept_id	Yes	integer	A foreign key that refers to a standard condition concept identifier in the Standardized Vocabularies.
condition_start_date	Yes	date	The date when the instance of the condition is recorded.
condition_end_date	No	date	The date when the instance of the condition is considered to have ended. If this information is not available, set to NULL.
condition_type_concept_id	Yes	integer	A foreign key to the predefined concept identifier in the Standardized Vocabularies reflecting the source data from which the condition was recorded, the level of standardization, and the type of occurrence. For example, conditions may be defined as primary or secondary diagnoses, problem lists and person statuses.
stop_reason	No	varchar(20)	The reason, if available, that the condition was no longer recorded, as indicated in the source data. Valid values include discharged, resolved, etc. Note that a stop_reason does not necessarily imply that the condition is no longer occurring.
provider_id	No	integer	A foreign key to the provider in the provider table who was responsible for determining (diagnosing) the condition.
visit_occurrence_id	No	integer	A foreign key to the visit in the visit table during which the condition was determined (diagnosed).

Field	Required	Type	Description
condition_source_value	No	varchar(50)	The source code for the condition as it appears in the source data. This code is mapped to a standard condition concept in the Standardized Vocabularies and the original code is, stored here for reference. Condition source codes are typically ICD-9-CM diagnosis codes from medical claims or discharge status/visit diagnosis codes from EHRs.
condition_source_concept_id	No	integer	A foreign key to a condition concept that refers to the code used in the source.

5.9.1 CONVENTIONS

- Valid Condition Concepts belong to the "Condition" domain. Standard Condition Concepts are based on SNOMED-CT.
- Condition records are typically inferred from diagnostic codes recorded in the source data. Such code system, like ICD-9-CM, ICD-10-CM, Read etc., provide a comprehensive coverage of conditions. However, if the code does not define a condition, but rather an observation or a procedure, then such information is not stored in the CONDITION_OCCURRENCE table, but in the respective tables instead.
- Source Condition identifiers are mapped to Standard Concepts for Conditions in the Standardized Vocabularies. When the source code cannot be translated into a Standard Concept, a CONDITION_OCCURRENCE entry is stored with only the corresponding source_concept_id (if available) and source_value and a condition_concept_id of 0.
- Family history and past diagnoses ("history of") are not recorded in the CONDITION_OCCURRENCE table. Instead, they are listed in the OBSERVATION table.
- Codes written in the process of establishing the diagnosis, such as "question of" of and "rule out", are not represented here. Instead, they are listed in the OBSERVATION table, if they are used for analyses.
- A Condition Occurrence Type is assigned based on the data source and type of condition attribute, including:
 - ICD-9-CM Primary Diagnosis from Inpatient and Outpatient Claims
 - ICD-9-CM Secondary Diagnoses from Inpatient and Outpatient Claims
 - Clinician diagnoses or problem Concepts from EHRs

5.10 MEASUREMENT

A measurement is the capture of a structured value (numerical or categorical) obtained through systematic examination of a person or sample. The Measurement domain captures measurement orders and measurement results. The measurement domain can contain laboratory results, vital signs, quantitative findings from pathology reports, etc.

Field	Required	Type	Description
measurement_id	Yes	integer	A unique identifier for each measurement.

Field	Required	Type	Description
person_id	Yes	integer	A foreign key identifier to the person about whom the measurement was recorded. The demographic details of that person are stored in the person table.
measurement_concept_id	Yes	integer	A foreign key to the standard measurement concept identifier in the Standardized Vocabularies.
measurement_date	Yes	date	The date of the Measurement.
measurement_time	No	time	The time of the Measurement
measurement_type_concept_id	Yes	integer	A foreign key to the predefined concept identifier in the Standardized Vocabularies reflecting the type of data on which the measurement record is based.
operator_concept_id	No	integer	A foreign key identifier to the mathematical operator that is applied to the value_as_number. Operators are <, ≤, =, ≥, >
value_as_number	No	float	A measurement stored as a number. This is applicable to measurement where the result is expressed as a numeric value.
value_as_concept_id	No	integer	A foreign key to a measurement stored as a concept identifier. This is applicable to measurements where the result can be expressed as a standard concept from the Standardized Vocabularies (e.g., positive/negative, present/absent, low/high, etc.).
unit_concept_id	No	integer	A foreign key to a standard concept identifier of measurement units in the Standardized Vocabularies.
range_low	No	float	The lower limit of the normal range of the measurement. The lower range is assumed to be in the same units of measure as the measurement value.
range_high	No	float	The upper limit of the normal range of the measurement. The lower range is assumed to be in the same units of measure as the measurement value.
provider_id	No	integer	A foreign key to the provider in the provider table who was responsible for making the measurement.
visit_occurrence_id	No	integer	A foreign key to the visit in the visit table during which the measurement was recorded.
measurement_source_value	No	varchar(50)	The measurement name as it appears in the source data. This code is mapped to a standard concept in the Standardized Vocabularies and the original code is, stored here for reference.
measurement_source_concept_id	No	integer	A foreign key to a concept that refers to the code used in the source.

Field	Required	Type	Description
unit_source_value	No	varchar(50)	The source code for the unit as it appears in the source data. This code is mapped to a standard unit concept in the Standardized Vocabularies and the original code is, stored here for reference.
value_source_value	No	varchar(50)	The source value associated with the structured value stored as numeric or concept. This field can be used in instances where the source data are transformed to produce the structured value.

5.10.1 CONVENTIONS

- Valid Measurement Concepts for both the measure (measurement_concept_id) and the measure result (value_as_concept) belong to the "Observation" domain. Measurement Concepts are based mostly on the LOINC vocabulary, with some additions from SNOMED-CT.
- Measurements are stored as attribute value pairs, where the attribute is the measure and the value represents the result. The value can be a concept (stored in value_as_concept), or a numerical value (value_as_number). The availability of a result is not mandatory.
- If reference ranges for upper and lower limit of normal as provided (typically by a laboratory) are stored in the range_high and range_low fields. Ranges have the same unit as the value_as_number.
- The Visit during which the observation was made is recorded through a reference to the VISIT_OCCURRENCE table. This information is not always available.
- The Provider making the observation is recorded through a reference to the PROVIDER table. This information is not always available.

5.11 NOTE

The note domain captures unstructured information that was recorded by a provider for a patient in free text notes on a given date.

Field	Required	Type	Description
note_id	Yes	integer	A unique identifier for each note.
person_id	Yes	integer	A foreign key identifier to the person about whom the note was recorded. The demographic details of that person are stored in the person table.
note_date	Yes	date	The date the note was recorded.
note_time	No	time	The time the note was recorded.
note_type_concept_id	Yes	integer	A foreign key to the predefined concept identifier in the Standardized Vocabularies reflecting the type data from which the note.
note_text	Yes	CLOB	The content of the note.
provider_id	No	integer	A foreign key to the provider in the provider table who was responsible for taking the note.
note_source_value	No	varchar(50)	The source value associated with the origin of the note, as standardized using the note_concept_id
visit_occurrence_id	No	integer	Foreign key to visit

5.11.1 CONVENTIONS

- The note table contains free text in UTF8 format taken by a healthcare provider.
- The Visit during which the note was written is recorded through a reference to the VISIT_OCCURRENCE table. This information is not always available.
- The Provider making the note is recorded through a reference to the PROVIDER table. This information is not always available.

5.12 OBSERVATION

The observation domain captures any clinical facts about a patient obtained in the context of examination, questioning or a procedure. The observation domain supports capture of data not represented by other domains, including unstructured measurements, medical history and family history.

Field	Required	Type	Description
observation_id	Yes	integer	A unique identifier for each observation.
person_id	Yes	integer	A foreign key identifier to the person about whom the observation was recorded. The demographic details of that person are stored in the person table.
observation_concept_id	Yes	integer	A foreign key to the standard observation concept identifier in the Standardized Vocabularies.
observation_date	Yes	date	The date of the observation.
observation_time	No	time	The time of the observation.
observation_type_concept_id	Yes	integer	A foreign key to the predefined concept identifier in the Standardized Vocabularies reflecting the type of the observation.
value_as_number	No	float	The observation result stored as a number. This is applicable to observations where the result is expressed as a numeric value.
value_as_string	No	varchar(60)	The observation result stored as a string. This is applicable to observations where the result is expressed as verbatim text.
value_as_concept_id	No	Integer	A foreign key to an observation result stored as a concept identifier. This is applicable to observations where the result can be expressed as a standard concept from the Standardized Vocabularies (e.g., positive/negative, present/absent, low/high, etc.).
qualifier_concept_id	No	integer	A foreign key to a standard concept identifier for a qualifier (e.g., severity of drug-drug interaction alert)
unit_concept_id	No	integer	A foreign key to a standard concept identifier of measurement units in the Standardized Vocabularies.

Field	Required	Type	Description
provider_id	No	integer	A foreign key to the provider in the provider table who was responsible for making the observation.
visit_occurrence_id	No	integer	A foreign key to the visit in the visit table during which the observation was recorded.
observation_source_value	No	varchar(50)	The observation code as it appears in the source data. This code is mapped to a standard concept in the Standardized Vocabularies and the original code is, stored here for reference.
observation_source_concept_id	No	integer	A foreign key to a concept that refers to the code used in the source.
unit_source_value	No	varchar(50)	The source code for the unit as it appears in the source data. This code is mapped to a standard unit concept in the Standardized Vocabularies and the original code is, stored here for reference.
qualifier_source_value	No	varchar(50)	The source value associated with a qualifier to characterize the observation

5.12.1 CONVENTIONS

- Valid Observation Concepts for the object (observation_concept_id) belong to the "Observation" domain. Observation Concepts are based mostly on the LOINC vocabulary, with some additions from SNOMED-CT.
- Valid Observation Concepts and the finding (value_as_concept_id) are not enforced by a domain but should be Standard Concepts.
- Observations must have an object represented as a concept, and a finding, represented as a concept, a numerical value or a verbatim string. There should be no observations records without an associated value. Observations which appear to be suggestive statements of positive assertion should have a recorded value as concept of 'Yes'.
- Observations obtained using standardized methods (e.g. laboratory assays) that produce discrete results are recorded by preference in the MEASUREMENT table.
- The Visit during which the observation was made is recorded through a reference to the VISIT_OCCURRENCE table. This information is not always available.
- The Provider making the observation is recorded through a reference to the PROVIDER table. This information is not always available.

5.13 FACT_RELATIONSHIP

The fact relationship table contains records to detail the relationships between facts within one domain or across two domains, and the nature of the relationship. Examples of types of fact relationships include: person relationships (mother-child linkage), care site relationships (representing the hierarchical organization structure of facilities within health systems), drug exposures provided due to associated indicated condition, devices used during the course of an associated procedure, and measurements derived from an associated specimen. All relationships are directional, and each relationship is represented twice symmetrically within the fact relationship table. For example, two persons

(PERSON_ID = 1 is the mother of PERSON_ID = 2) have two fact relationships: 1- 'PERSON_ID 1' 'parent of' 'PERSON_ID 2', and 2-'PERSON_ID 2' 'child of' 'PERSON_ID 1'.

Field	Required	Type	Description
domain_concept_id_1	Yes	integer	The concept representing the domain of fact one, from which the corresponding table can be inferred.
fact_id_1	Yes	integer	The unique identifier in the table corresponding to the domain of fact one.
domain_concept_id_2	Yes	integer	The concept representing the domain of fact two, from which the corresponding table can be inferred.
fact_id_2	Yes	integer	The unique identifier in the table corresponding to the domain of fact two.
relationship_concept_id	Yes	integer	A foreign key to a standard concept identifier of relationship in the Standardized Vocabularies.

5.13.1 CONVENTIONS

- For each pair between fact one and fact two should exist two records, one in each direction. For example, if fact one is related to person two, then the first record should contain the 'is a' in fact_id_1 and the second fact in fact_id_2, and the relationship_concept_id for "is a". The second record should contain act one in fact_id_2 and fact two in fact_id_1 and the relationship_concept_id for "is a".

6 Standardized Health System Data Tables

These tables describe the healthcare provider system responsible for administering the healthcare of the patient, rather than the demographic or clinical events the patient is involved in.

Below provides an entity-relationship diagram highlighting the tables within the Standardized Health System portion of the OMOP Common Data Model:



Standardized Health System Data Entity-Relationship diagram

6.1 LOCATION

The Location table represents a generic way to capture physical location or address information. Locations are used to define the addresses for Persons and Care Sites.

Field	Required	Type	Description
location_id	Yes	integer	A unique identifier for each geographic location.
address_1	No	varchar(50)	The address field 1, typically used for the street address, as it appears in the source data.
address_2	No	varchar(50)	The address field 2, typically used for additional detail such as buildings, suites, floors, as it appears in the source data.
city	No	varchar(50)	The city field as it appears in the source data.
state	No	varchar(2)	The state field as it appears in the source data.
zip	No	varchar(9)	The zip or postal code. For US addresses, valid zip codes can be 3, 5 or 9 digits long, depending on the source data.
county	No	varchar(20)	The county. The county information is necessary because not all zip codes fall into one and the same county.
location_source_value	No	varchar(50)	The verbatim information that is used to uniquely identify the location as it appears in the source data.

6.1.1 CONVENTIONS

- Each address or Location is unique and is present only once in the table.
- Locations do not contain names. In order to construct a full address that can be used on the Postal Service, the address information from the Location needs to be combined with information from the Care Site. The Person table does not contain name information.
- All fields in the Location tables contain the verbatim data in the Source. None of them are mandatory, but a valid Location record should at least contain either a Location Name or Location Zip.
- Zip codes are handled as strings of up to 9 characters length. For US addresses, these represent either a 3-digit abbreviated Zip code as provided by many Sources for Patient protection reasons, or the full 5-digit Zip code or the 9-digit (ZIP + 4) codes are recorded. Unless for specific reasons, analytical methods should expect and utilize only the first 3 digits. For international addresses, different rules apply.

6.2 CARE_SITE

The Care Site table contains a list of uniquely identified physical or organizational units where healthcare delivery is practiced (offices, wards, hospitals, clinics, etc.).

Field	Required	Type	Description
care_site_id	Yes	Integer	A unique identifier for each organization. Here, an organization is defined as a collection of one or more care sites that share a single EHR database.

Field	Required	Type	Description
care_site_name	No	varchar(255)	The description of the care site
place_of_service_concept_id	No	Integer	A foreign key that refers to a place of service concept identifier in the Standardized Vocabularies.
location_id	No	Integer	A foreign key to the geographic location of the administrative offices of the organization in the location table, where the detailed address information is stored.
care_site_source_value	No	varchar(50)	The identifier for the organization in the source data, stored here for reference.
place_of_service_source_value	No	varchar(50)	The source code for the place of service as it appears in the source data, stored here for reference.

6.2.1 CONVENTIONS

- There can be hierarchical and business relationships between Care Sites (e.g., wards can belong to clinics, which can in turn belong to hospitals, which in turn can belong to hospital systems, which in turn can belong to HMOs). These relationships should be defined in the FACT_RELATIONSHIP table.
- The Care Site Source Value typically contains the name of the Care Site.
- The Place of Service Concepts belongs to the Domain "Provider". These Concepts are based on a catalog maintained by the CMS

6.3 PROVIDER

The Provider table contains a list of uniquely identified health care providers. These are typically physicians, nurses, etc.

Field	Required	Type	Description
provider_id	Yes	Integer	A unique identifier for each provider.
provider_name	No	varchar(50)	A description of the provider
npi	No	varchar(20)	The National Provider Identifier (NPI) of the provider.
dea	No	varchar(20)	The Drug Enforcement Administration (DEA) number of the provider.
specialty_concept_id	No	Integer	A foreign key to a standard provider's specialty concept identifier in the Standardized Vocabularies.
care_site_id	No	Integer	A foreign key to the main care site where the provider is practicing.
year_of_birth	No	Integer	
gender_concept_id	No	Integer	
provider_source_value	No	varchar(50)	The identifier used for the provider in the source data, stored here for reference.

Field	Required	Type	Description
specialty_source_value	No	varchar(50)	The source code for the provider specialty as it appears in the source data, stored here for reference.
specialty_source_concept_id	No	integer	A foreign key to a concept that refers to the code used in the source.
gender_source_value	No	varchar(50)	
gender_source_concept_id	No	integer	A foreign key to a concept that refers to the code used in the source.

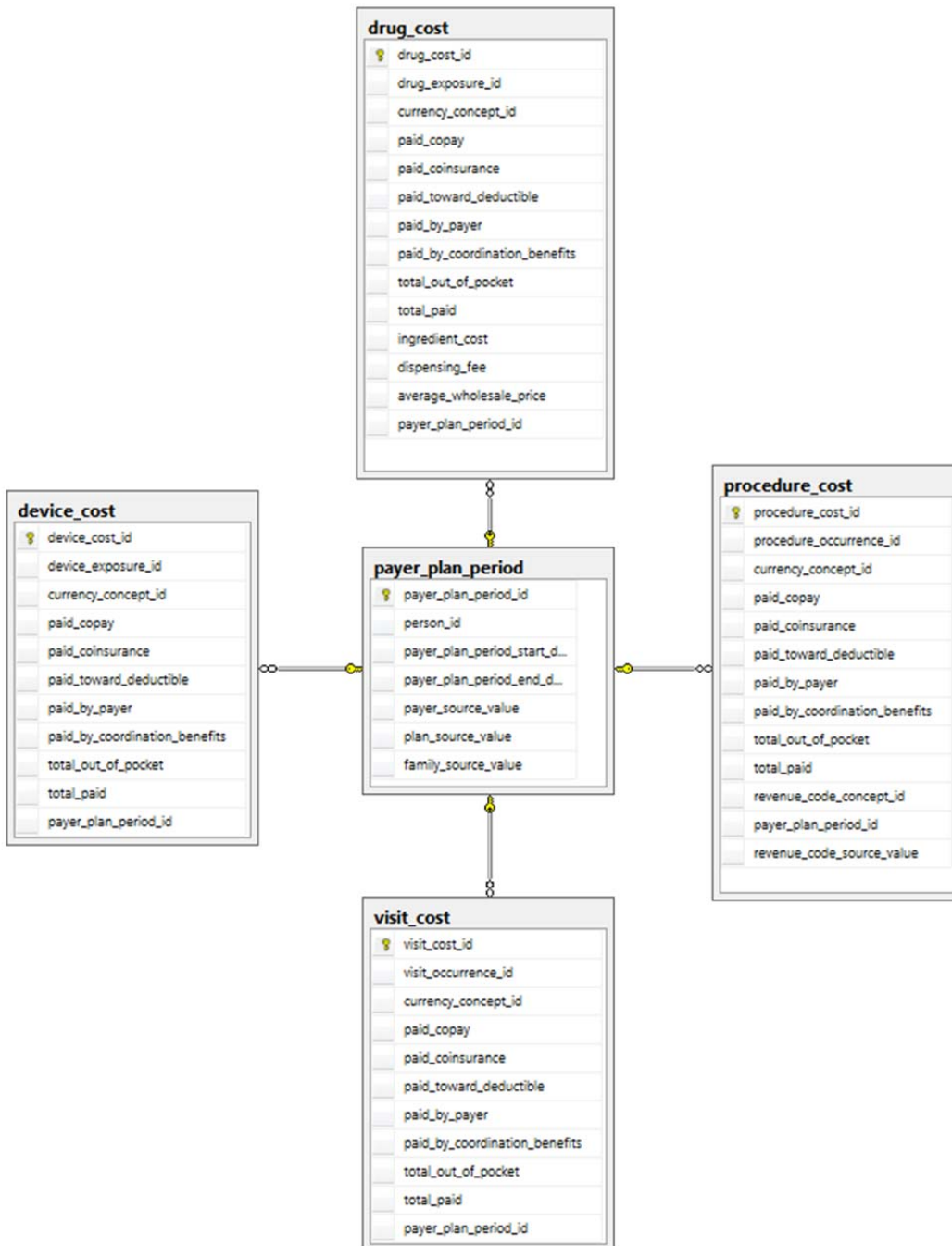
6.3.1 CONVENTIONS

- Providers are not duplicated in the table.
- Valid Specialty Concepts for both the test (measurement_concept_id) belong to the "Provider" domain. The Specialty Concepts are based on the CDC specialty classification.
- This table is used to represent fixed relationship between Providers and Care Sites. Providers are also linked to Care Sites through Condition, Procedure and Visit records.

7 Standardized Health Economics Data Tables

These tables contain cost information about healthcare. They are dependent on the healthcare delivery system the patient population is involved in, which may vary significantly across different countries. However, the current model is focused on the US healthcare system.

Below provides an entity-relationship diagram highlighting the tables within the Standardized Health Economics portion of the OMOP Common Data Model:



7.1 PAYER_PLAN_PERIOD

The payer plan period domain captures records that detail the period of time that a person is continuously enrolled under a specific health plan benefit structure from a given payer. Each Person receiving health care and covered by a health benefits is subject to a Plan defined by the Payer for the Person or her family. For a given benefit policy, there may be one or more Plans that are active for certain periods of time (e.g. before and after the deductible is reached), determining the cost of health services provided.

Field	Required	Type	Description
payer_plan_period_id	Yes	integer	An identifier for each unique combination of payer, plan, family code and time span.
person_id	Yes	integer	A foreign key identifier to the person covered by the payer. The demographic details of that person are stored in the person table.
payer_plan_period_start_date	Yes	date	The start date of the payer plan period.
payer_plan_period_end_date	Yes	date	The end date of the payer plan period.
payer_source_value	No	varchar(50)	The source code for the payer as it appears in the source data.
plan_source_value	No	varchar(50)	The source code for the person's coverage plan as it appears in the source data.
family_source_value	No	varchar(50)	The source code for the person's family as it appears in the source data.

7.1.1 CONVENTIONS

Different Payers have different designs for their health benefit Plans. The Payer Plan Period table does not capture all details of the plan design or the relationship between the Plan and the cost of healthcare. However, it allows identifying the unique combination of Payer (insurer), Plan (determining health care benefits and limits) and Family membership for each Person. Usually, depending on health care utilization a Person may have one or many subsequent Plans during coverage by a single Payer. The tables captures the period a plan is active (Start and End Date), the Payer Source Value (name or ID of the Payer), the Plan Source Value (name or ID of the Plan) and Family Source Value (ID of the family, which may consist of just one family member – the Person).

7.2 VISIT_COST

The Visit Cost table captures the costs of health visit of a patient which are not itemized to specific procedures, drugs, or devices used within the encounter.

Field	Required	Type	Description
visit_cost_id	Yes	integer	A unique identifier for each procedure cost record.
visit_occurrence_id	Yes	integer	A foreign key identifier to the procedure record for which cost data are recorded.
currency_concept_id	No	integer	A concept representing the 3-letter code used to delineate international currencies, such as USD for

Field	Required	Type	Description
			US Dollar.
paid_copay	No	float	The amount paid by the person as a fixed contribution to the expenses. Copay does not contribute to the out_of_pocket expenses.
paid_coinsurance	No	float	The amount paid by the person as a joint assumption of risk. Typically, this is a percentage of the expenses defined by the payer plan (policy) after the person's deductible is exceeded.
paid_toward_deductible	No	float	The amount paid by the person that is counted toward the deductible defined by the payer plan (policy).
paid_by_payer	No	float	The amount paid by the payer (insurer). If there is more than one payer, several visit_cost records indicate that fact.
paid_by_coordination_benefits	No	float	The amount paid by a secondary payer through the coordination of benefits.
total_out_of_pocket	No	float	The total amount paid by the person as a share of the expenses, excluding the copay.
total_paid	No	float	The total amount paid for the expenses of the procedure.
payer_plan_period_id	No	integer	A foreign key to the payer_plan_period table, where the details of the payer, plan and family are stored.

7.2.1 CONVENTIONS

- The cost of the visit may contain just board and food, but could also include the entire cost of everything that was happening to the patient during the visit.
- All other conventions apply as in the PROCEDURE_COST table.

7.3 PROCEDURE_COST

The procedure cost table captures the cost of a Procedure performed on a Person. The information about the cost is only derived from the amounts paid for the Procedure. This is in contrast to the Drug Cost data which also contain information about true amount charged by the distributor. In addition, Revenue codes are captured.

Field	Required	Type	Description
procedure_cost_id	Yes	integer	A unique identifier for each procedure cost record.
procedure_occurrence_id	Yes	integer	A foreign key identifier to the procedure record for which cost data are recorded.

Field	Required	Type	Description
currency_concept_id	No	integer	A concept representing the 3-letter code used to delineate international currencies, such as USD for US Dollar.
paid_copay	No	float	The amount paid by the person as a fixed contribution to the expenses. Copay does not contribute to the out_of_pocket expenses.
paid_coinsurance	No	float	The amount paid by the person as a joint assumption of risk. Typically, this is a percentage of the expenses defined by the payer plan (policy) after the person's deductible is exceeded.
paid_toward_deductible	No	float	The amount paid by the person that is counted toward the deductible defined by the payer plan (policy).
paid_by_payer	No	float	The amount paid by the payer (insurer). If there is more than one payer, several procedure_cost records indicate that fact.
paid_by_coordination_benefits	No	float	The amount paid by a secondary payer through the coordination of benefits.
total_out_of_pocket	No	float	The total amount paid by the person as a share of the expenses
total_paid	No	float	The total amount paid for the expenses of the procedure.
revenue_code_concept_id	No	integer	A foreign key referring to a standard concept identifier in the Standardized Vocabularies for revenue codes.
payer_plan_period_id	No	integer	A foreign key to the payer_plan_period table, where the details of the payer, plan and family are stored.
revenue_code_source_value	No	varchar(50)	The source code for the revenue code as it appears in the source data, stored here for reference.

7.3.1 CONVENTIONS

- Each Procedure Occurrence may have any number of corresponding records in the Procedure Cost table, but typically it is none (cost data not captured) or one (one payment per Procedure). They are linked directly through the Procedure Occurrence ID field.

The amounts paid are:

- Copay – a fixed amount to be paid by the Person
- Coinsurance – a relative amount of the total paid by the Person
- Deductible – an amount of money paid by the Person before the Payer starts contributing
- Primary Payer – the amount the primary Payer pays towards the total
- Coordination of Benefits – the amount a secondary Payer or Family Plan pays towards the total
- Out of Pocket = Copay + Coinsurance + Deductible
- Total – the total amount paid for the procedure

The amounts in various payment components should equal the total, so Copay + Coinsurance + Deductible + Primary Payer + COB = Total Paid. In reality, this is not always reflected in the source data. It is up to the ETL to determine how to deal with quality problems in the data.

There are important indicators for the amount paid that are determined through the health plan design:

- Revenue Codes – determining what service within a provider is charging for the service

All these data are captured as Source Values and Concept IDs referring to the Standardized Vocabularies.

Finally, the health plan of the Person that is determined by these numbers is referred to through the Payer Plan Period ID (see below).

7.4 DRUG_COST

The Drug Cost table captures records indicating the cost of a Drug Exposure. The information about the cost is defined by the amount of money paid by the person and payer for the drug, as well as the charged cost of the drug.

In addition, a reference to the health plan information in the Payer Plan Period table is stored in the record that is responsible for the determination of the cost as well as some of the payments.

Field	Required	Type	Description
drug_cost_id	Yes	integer	A unique identifier for each drug cost record.
drug_exposure_id	Yes	integer	A foreign key identifier to the drug record for which cost data are recorded.
currency_concept_id	No	integer	A concept representing the 3-letter code used to delineate international currencies, such as USD for US Dollar.
paid_copay	No	float	The amount paid by the person as a fixed contribution to the expenses. Copay does not contribute to the out of pocket expenses.
paid_coinsurance	No	float	The amount paid by the person as a joint assumption of risk. Typically, this is a percentage of the expenses defined by the payer plan (policy) after the person's deductible is exceeded.
paid_toward_deductible	No	float	The amount paid by the person that is counted toward the deductible defined by the payer plan (policy).
paid_by_payer	No	float	The amount paid by the payer (insurer). If there is more than one payer, several drug_cost records indicate that fact.
paid_by_coordination_benefits	No	float	The amount paid by a secondary payer through the coordination of benefits.
total_out_of_pocket	No	float	The total amount paid by the person as a share of the expenses
total_paid	No	float	The total amount paid for the expenses of drug exposure.
ingredient_cost	No	float	The portion of the drug expenses due to the cost charged by the manufacturer for the drug, typically a percentage of the Average Wholesale Price.

Field	Required	Type	Description
dispensing_fee	No	float	The portion of the drug expenses due to the dispensing fee charged by the pharmacy, typically a fixed amount.
average_wholesale_price	No	float	List price of a drug set by the manufacturer.
payer_plan_period_id	No	integer	A foreign key to the payer_plan_period table, where the details of the payer, plan and family are stored.

7.4.1 CONVENTIONS

- Each Drug Exposure may have any number of corresponding records in the DRUG COST table, but typically it is none (no cost data recorded) or one. They are linked directly through the drug_exposure_id field.

The amounts paid are:

- Copay – a fixed amount to be paid by the Person
- Coinsurance – a relative amount of the total paid by the Person
- Deductible – an amount of money paid by the Person before the Payer starts contributing
- Primary Payer – the amount the primary Payer pays towards the total
- Coordination of Benefits – the amount a secondary Payer or Family Plan pays towards the total
- Out of Pocket = Copay + Coinsurance + Deductible
- Total – the total amount paid for the Drug Exposure

The costs are:

- Ingredient Cost – the amount charged by the wholesale distributor or manufacturer
- Dispensing Fee – the amount charged by the pharmacy
- Sales Tax. This is usually very small and typically not provided by most source data, and therefore not included in the CDM

The amount paid should equal the cost, so Copay + Coinsurance + Deductible + Primary Payer + Coordination of Benefits = Total Paid = Ingredient Cost + Dispensing Fee. In reality, this is not always reflected in the source data. It is up to the ETL to determine how to deal with quality problems in the data.

The Average Wholesale Price is the list price of the drug, but not the price actually charged or paid.

Finally, the health plan of the Person that is determined by these numbers is referred to through the Payer Plan Period ID (see below).

7.5 DEVICE_COST

The Device Cost table captures the cost of a medical device or supply used on a Person. The information about the cost is only derived from the amounts paid for the device.

Field	Required	Type	Description
device_cost_id	Yes	integer	A unique identifier for each procedure cost record.
device_exposure_id	Yes	integer	A foreign key identifier to the procedure record for which cost data are recorded.

Field	Required	Type	Description
currency_concept_id	No	integer	A concept representing the 3-letter code used to delineate international currencies, such as USD for US Dollar.
paid_copay	No	float	The amount paid by the person as a fixed contribution to the expenses. Copay does not contribute to the out_of_pocket expenses.
paid_coinsurance	No	float	The amount paid by the person as a joint assumption of risk. Typically, this is a percentage of the expenses defined by the payer plan (policy) after the person's deductible is exceeded.
paid_toward_deductible	No	float	The amount paid by the person that is counted toward the deductible defined by the payer plan (policy).
paid_by_payer	No	float	The amount paid by the payer (insurer). If there is more than one payer, several procedure_cost records indicate that fact.
paid_by_coordination_benefits	No	float	The amount paid by a secondary payer through the coordination of benefits.
total_out_of_pocket	No	float	The total amount paid by the person as a share of the expenses, excluding the copay.
total_paid	No	float	The total amount paid for the expenses of the procedure.
payer_plan_period_id	No	integer	A foreign key to the payer_plan_period table, where the details of the payer, plan and family are stored.

7.5.1 CONVENTIONS

- If the device is derived from a procedure record recorded as a HCPCS or CPT-4, all conventions apply to the field's equivalent to the procedure_cost (see above).

8 Standardized Derived Elements

These tables contain information about the clinical events of a patient that is not derived from the raw source data, but from other tables of the CDM.

Below provides an entity-relationship diagram highlighting the tables within the Standardized Derived Elements portion of the OMOP Common Data Model:



8.1 COHORT

The Cohort table contains records derived as the collection of subjects that satisfy a given set of inclusion criteria for a duration of time. The definition of the cohort is contained within the cohort definition table. Example cohorts can include patients diagnosed with a specific condition, patients exposed to a particular drug, or providers who have performed a specific procedure.

Field	Required	Type	Description
cohort_definition_id	Yes	integer	A foreign key to a record in the COHORT_DEFINITION table containing relevant Cohort definition information.
subject_id	Yes	integer	A foreign key to the subject in the cohort. These could be referring to records in the Person, Provider, Visit Occurrence table.
cohort_start_date	Yes	date	The date when the cohort definition criteria for the person, provider or visit first match.
cohort_end_date	Yes	date	The date when the cohort definition criteria for the person, provider or visit no longer match or the cohort membership was terminated.

8.1.1 CONVENTIONS

- The core of a Cohort is the definition of the unifying definition or feature of the Cohort. This is captured in the cohort_definition_id.
- Cohort records must have a Start Date
- Cohort records must have an End Date, but may be set to Start Date or could apply a censor date using the Observation Period Start Date.
- Cohort records must contain a Subject ID, which can refer to a Person, Provider, or Visit record. The cohort definition will define the type of subject through the subject concept id.

8.2 COHORT_ATTRIBUTE

The Cohort Attribute table contains attributes associated with each subject within a cohort, as defined by a given set of inclusion criteria for a duration of time. The definition of the cohort attribute is contained within the attribute definition table. Example cohort attributes can be age, BMI or comorbidity score.

Field	Required	Type	Description
cohort_definition_id	Yes	integer	A foreign key to a record in the COHORT_DEFINITION table containing relevant Cohort definition information.
subject_id	Yes	integer	A foreign key to the subject in the cohort. These could be referring to records in the PERSON, PROVIDER, or VISIT_OCCURRENCE table.
cohort_start_date	Yes	date	The date when the cohort definition criteria for the person, provider or visit first match.

Field	Required	Type	Description
cohort_end_date	Yes	date	The date when the cohort definition criteria for the person, provider or visit no longer match or the cohort membership was terminated.
attribute_definition_id	Yes	integer	A foreign key to a record in the ATTRIBUTE_DEFINITION table containing relevant attribute definition information.
value_as_number	No	float	The attribute result stored as a number. This is applicable to attributes where the result is expressed as a numeric value, such as calculated entities (e.g. age, BMI) or composite scales (e.g. Charlson index).
value_as_concept_id	No	integer	The attribute result stored as a concept id. This is applicable to attributes where the result is expressed as a categorical value.

8.2.1 CONVENTIONS

- The core of a Cohort is the definition of the unifying definition or feature of the Cohort. This is captured in the cohort_definition_id.
- Cohort records must have a Start Date
- Cohort records must have an End Date, but may be set to Start Date or could apply a censor date using the Observation Period Start Date.
- Cohort records must contain a Subject ID, which can refer to a Person, Provider, or Visit record. The cohort definition will define the type of subject through the subject concept id.

8.3 DRUG_ERA

A Drug Era is defined as a span of time when the Person is assumed to be exposed to a particular active ingredient. A Drug Era is not the same as a Drug Exposure: Exposures are individual records corresponding to the source when drug was delivered to the Person, while successive periods of Drug Exposures are combined under certain rules to produce continuous Drug Eras.

Field	Required	Type	Description
drug_era_id	Yes	integer	A unique identifier for each drug era.
person_id	Yes	integer	A foreign key identifier to the person who is subjected to the drug during the drug era. The demographic details of that person are stored in the person table.
drug_concept_id	Yes	integer	A foreign key that refers to a standard concept identifier in the Standardized Vocabularies for the drug concept.
drug_era_start_date	Yes	date	The start date for the drug era constructed from the individual instances of drug exposures. It is the start date of the very first chronologically recorded instance of utilization of a drug.
drug_era_end_date	Yes	date	The end date for the drug era constructed from the individual instance of drug exposures. It is the end date of the final continuously recorded instance of utilization of a drug.

Field	Required	Type	Description
drug_exposure_count	No	integer	The number of individual drug exposure occurrences used to construct the drug era.
gap_days	No	integer	The number of observed days of gap between drug exposure records which was tolerated in the building of the drug era record.

8.3.1 CONVENTIONS

- Drug Eras will be derived from DRUG_EXPOSURE using a standardized algorithm.
- Each Drug Era corresponds to one or many Drug Exposures that form a continuous interval and contain the same drug ingredient (active compound).
- The drug_concept_id field contains Concepts that have the concept_class "Ingredient". The ingredient is derived from the Drugs in the DRUG_EXPOSURE table that are aggregated into the Drug Era record.
- The Drug Era Start Date is the start date of the first Drug Exposure.
- The Drug Era End Date is the end date of the last Drug Exposure.
- The End Date of each Drug Exposure is either taken from the field Drug Exposure End Date or, as it is typically not available, inferred using the following rules:
 - For pharmacy prescriptions claims, the date when the drug was dispensed and the number of days of supply are used to extrapolate the End Date for the Drug Exposure.
 - For Procedure Drugs, usually the drug is administered on a single date (i.e., the administration date).

8.4 DOSE_ERA

A Dose Era is defined as a span of time when the Person is assumed to be exposed to a constant dose of a specific active ingredient.

Field	Required	Type	Description
dose_era_id	Yes	integer	A unique identifier for each drug era.
person_id	Yes	integer	A foreign key identifier to the person who is subjected to the drug during the drug era. The demographic details of that person are stored in the person table.
drug_concept_id	Yes	integer	A foreign key that refers to a standard concept identifier in the Standardized Vocabularies for the active ingredient drug concept.
unit_concept_id	Yes	integer	A foreign key that refers to a standard concept identifier in the Standardized Vocabularies for the unit concept.
dose_value	Yes	float	The numeric value of the dose
dose_era_start_date	Yes	date	The start date for the drug era constructed from the individual instances of drug exposures. It is the start date of the very first chronologically recorded instance of utilization of a drug.

Field	Required	Type	Description
dose_era_end_date	Yes	date	The end date for the drug era constructed from the individual instance of drug exposures. It is the end date of the final continuously recorded instance of utilization of a drug.

8.4.1 CONVENTIONS

- Dose Eras will be derived from DRUG_EXPOSURE using a standardized algorithm.
- Each Dose Era corresponds to one or many Drug Exposures that form a continuous interval and contain the same drug ingredient (active compound) at the same effective dose.

8.5 CONDITION_ERA

A Condition Era is defined as a span of time when the Person is assumed to have a given condition.

Similar to Drug Eras, Condition Eras are chronological periods of Condition Occurrence. Combining individual Condition Occurrences into a single Condition Era serves two purposes:

- It allows aggregation of chronic conditions that require frequent ongoing care, instead of treating each Condition Occurrence as an independent event.
- It allows aggregation of multiple, closely timed doctor visits for the same condition to avoid double-counting the Condition Occurrences.

For example, consider a Person who visits her Primary Care Physician (PCP) and who is diagnosed leading to a referral to a specialist. One week later, the Person visits the specialist, who confirms the PCP's diagnosis and provides the appropriate treatment to resolve the condition. These two independent doctor visits should be aggregated into one Condition Era.

Field	Required	Type	Description
condition_era_id	Yes	integer	A unique identifier for each condition era.
person_id	Yes	integer	A foreign key identifier to the person who is experiencing the condition during the condition era. The demographic details of that person are stored in the person table.
condition_concept_id	Yes	integer	A foreign key that refers to a standard condition concept identifier in the Standardized Vocabularies.
condition_era_start_date	Yes	date	The start date for the condition era constructed from the individual instances of condition occurrences. It is the start date of the very first chronologically recorded instance of the condition.
condition_era_end_date	Yes	date	The end date for the condition era constructed from the individual instances of condition occurrences. It is the end date of the final continuously recorded instance of the condition.
condition_occurrence_count	No	integer	The number of individual condition occurrences used to construct the condition era.

8.5.1 CONVENTIONS

- Condition Era records will be derived from the `CONDITION_OCCURRENCE` table using a standardized algorithm.
- Each Condition Era corresponds to one or many `CONDITION_OCCURRENCE` records that form a continuous interval and contain the same drug ingredient (active compound).
- The `condition_concept_id` field contains Concepts that are identical to those of the `CONDITION_OCCURRENCE` table records that make up the Condition Era.
- The Condition Era Start Date is the start date of the first Condition Occurrence.
- The Condition Era End Date is the end date of the last Condition Occurrence.