# **OMOP CDM v5.4**

This is the specification document for the OMOP Common Data Model, v5.4. **This is the latest version of the OMOP CDM.** Each table is represented with a high-level description and ETL conventions that should be followed. This is continued with a discussion of each field in each table, any conventions related to the field, and constraints that should be followed (like primary key, foreign key, etc). All tables should be instantiated in a CDM instance but do not need to be populated. Similarly, fields that are not required should exist in the CDM table but do not need to be populated. Should you have questions please feel free to visit the [forums](https://forums.ohdsi.org/) or the [github issue](https://github.com/ohdsi/CommonDataModel/issues) page.

### Current Support for CDM v5.4

The table below details which OHDSI tools support CDM v5.4. There are two levels of support: legacy support means that the tool supports all tables and fields that were present in CDM v5.3 and feature support indicates that the tool supports any new tables and fields in CDM v5.4 that were not present in CDM v5.3. A green check ✔️ indicates that the support level for the listed tool is in place, has been tested, and released. A warning sign ❗ indicates that the support level for the listed tool has been initiated but has not yet been tested and released.

| **Tool** | **Description** | **Legacy Support** | **Feature Support** |
| --- | --- | --- | --- |
| **CDM R package** | This package can be downloaded from <https://github.com/OHDSI/CommonDataModel/>. It functions to dynamically create the OMOP CDM documentation and DDL scripts to instantiate the CDM tables. | ✔️ | ✔️ |
| **Data Quality Dashboard** | This package can be downloaded from <https://github.com/OHDSI/DataQualityDashboard>. It runs a set of > 3500 data quality checks against an OMOP CDM instance and is required to be run on all databases prior to participating in an OHDSI network research study. | ✔️ | ❗ |
| **Achilles** | This package can be downloaded from <https://github.com/OHDSI/Achilles>, performing a set of broad database characterizations against an OMOP CDM instance. | ✔️ | ❗ |
| **ARES** | This package can be downloaded from <https://github.com/OHDSI/Ares> and is designed to display the results from both the ACHILLES and DataQualityDashboard packages to support data quality and characterization research. | ✔️ | ❗ |
| **ATLAS** | ATLAS is an open source software tool for researchers to conduct scientific analyses on standardized observational data. [Demo](http://atlas-demo.ohdsi.org/) | ✔️ | ❗ |
| **Rabbit-In-A-Hat** | This package can be downloaded from <https://github.com/OHDSI/WhiteRabbit> and is an application for interactive design of an ETL to the OMOP Common Data Model with the help of the the scan report generated by White Rabbit. | ✔️ | ✔️ |
| **Feature Extraction** | This package can be downloaded from <https://github.com/OHDSI/FeatureExtraction>. It is designed to generate features (covariates) for a cohort generated using the OMOP CDM. | ✔️ | ✔️\* |
| **Cohort Diagnostics** | This package can be downloaded from <https://github.com/OHDSI/CohortDiagnostics> and is used to critically evaluate cohort phenotypes. | ✔️ | ❗ |

\* The **Feature Extraction** package supports all relevant new features in CDM v5.4. For example, it was decided that, from a methodological perspective, the EPISODE and EPISODE\_EVENT tables should not be included to define cohort covariates because the events that make up episodes are already pulled in as potential covariates.

Looking to send us a pull request for a bug fix? Please see the [readme](https://github.com/OHDSI/CommonDataModel#readme) on the main github page.

### person

**Table Description**

This table serves as the central identity management for all Persons in the database. It contains records that uniquely identify each person or patient, and some demographic information.

**User Guide**

All records in this table are independent Persons.

**ETL Conventions**

All Persons in a database needs one record in this table, unless they fail data quality requirements specified in the ETL. Persons with no Events should have a record nonetheless. If more than one data source contributes Events to the database, Persons must be reconciled, if possible, across the sources to create one single record per Person. The content of the BIRTH\_DATETIME must be equivalent to the content of BIRTH\_DAY, BIRTH\_MONTH and BIRTH\_YEAR.  
  
For detailed conventions for how to populate this table, please refer to the [THEMIS repository](https://ohdsi.github.io/Themis/person.html).

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **person\_id** | It is assumed that every person with a different unique identifier is in fact a different person and should be treated independently. | Any person linkage that needs to occur to uniquely identify Persons ought to be done prior to writing this table. This identifier can be the original id from the source data provided if it is an integer, otherwise it can be an autogenerated number. | integer | Yes | Yes | No |  |  |
| **gender\_concept\_id** | This field is meant to capture the biological sex at birth of the Person. This field should not be used to study gender identity issues. | Use the gender or sex value present in the data under the assumption that it is the biological sex at birth. If the source data captures gender identity it should be stored in the [OBSERVATION](https://ohdsi.github.io/CommonDataModel/cdm531.html#observation) table. [Accepted gender concepts](http://athena.ohdsi.org/search-terms/terms?domain=Gender&standardConcept=Standard&page=1&pageSize=15&query=). Please refer to the [THEMIS repository](https://ohdsi.github.io/Themis/tag_gender_concept_id.html) for detailed conventions on how to populate this field. | integer | Yes | No | Yes | CONCEPT | Gender |
| **year\_of\_birth** | Compute age using year\_of\_birth. | For data sources with date of birth, the year should be extracted. If no year of birth is available all the person’s data should be dropped from the CDM instance. For additional information on how to populate this field, please refer to the [THEMIS repository](https://ohdsi.github.io/Themis/tag_year_of_birth.html). | integer | Yes | No | No |  |  |
| **month\_of\_birth** |  | For data sources that provide the precise date of birth, the month should be extracted and stored in this field. | integer | No | No | No |  |  |
| **day\_of\_birth** |  | For data sources that provide the precise date of birth, the day should be extracted and stored in this field. | integer | No | No | No |  |  |
| **birth\_datetime** |  | This field is not required but highly encouraged. For data sources that provide the precise datetime of birth, that value should be stored in this field. For more information on how to populate this field, please refer to the [THEMIS repository](https://ohdsi.github.io/Themis/person.html). | datetime | No | No | No |  |  |
| **race\_concept\_id** | This field captures race or ethnic background of the person. | Only use this field if you have information about race or ethnic background. The Vocabulary contains Concepts about the main races and ethnic backgrounds in a hierarchical system. Due to the imprecise nature of human races and ethnic backgrounds, this is not a perfect system. Mixed races are not supported. If a clear race or ethnic background cannot be established, use Concept\_Id 0. [Accepted Race Concepts](http://athena.ohdsi.org/search-terms/terms?domain=Race&standardConcept=Standard&page=1&pageSize=15&query=). | integer | Yes | No | Yes | CONCEPT | Race |
| **ethnicity\_concept\_id** | This field captures Ethnicity as defined by the Office of Management and Budget (OMB) of the US Government: it distinguishes only between “Hispanic” and “Not Hispanic”. Races and ethnic backgrounds are not stored here. | Only use this field if you have US-based data and a source of this information. Do not attempt to infer Ethnicity from the race or ethnic background of the Person. [Accepted ethnicity concepts](http://athena.ohdsi.org/search-terms/terms?domain=Ethnicity&standardConcept=Standard&page=1&pageSize=15&query=) | integer | Yes | No | Yes | CONCEPT | Ethnicity |
| **location\_id** | The location refers to the physical address of the person. This field should capture the last known location of the person. | Put the location\_id from the [LOCATION](https://ohdsi.github.io/CommonDataModel/cdm531.html#location) table here that represents the most granular location information for the person. For additional information on how to populate this field, please refer to the [THEMIS repository](https://ohdsi.github.io/Themis/populate_person_location_id.html). | integer | No | No | Yes | LOCATION |  |
| **provider\_id** | The Provider refers to the last known primary care provider (General Practitioner). | Put the provider\_id from the [PROVIDER](https://ohdsi.github.io/CommonDataModel/cdm531.html#provider) table of the last known general practitioner of the person. If there are multiple providers, it is up to the ETL to decide which to put here. | integer | No | No | Yes | PROVIDER |  |
| **care\_site\_id** | The Care Site refers to where the Provider typically provides the primary care. |  | integer | No | No | Yes | CARE\_SITE |  |
| **person\_source\_value** | Use this field to link back to persons in the source data. This is typically used for error checking of ETL logic. | Some use cases require the ability to link back to persons in the source data. This field allows for the storing of the person value as it appears in the source. This field is not required but strongly recommended. | varchar(50) | No | No | No |  |  |
| **gender\_source\_value** | This field is used to store the biological sex of the person from the source data. It is not intended for use in standard analytics but for reference only. | Put the assigned sex at birth of the person as it appears in the source data. | varchar(50) | No | No | No |  |  |
| **gender\_source\_concept\_id** | Due to the small number of options, this tends to be zero. | If the source data codes asigned sex at birth in a non-standard vocabulary, store the concept\_id here. | integer | No | No | Yes | CONCEPT |  |
| **race\_source\_value** | This field is used to store the race of the person from the source data. It is not intended for use in standard analytics but for reference only. | Put the race of the person as it appears in the source data. | varchar(50) | No | No | No |  |  |
| **race\_source\_concept\_id** | Due to the small number of options, this tends to be zero. | If the source data codes race in an OMOP supported vocabulary store the concept\_id here. | integer | No | No | Yes | CONCEPT |  |
| **ethnicity\_source\_value** | This field is used to store the ethnicity of the person from the source data. It is not intended for use in standard analytics but for reference only. | If the person has an ethnicity other than the OMB standard of “Hispanic” or “Not Hispanic” store that value from the source data here. | varchar(50) | No | No | No |  |  |
| **ethnicity\_source\_concept\_id** | Due to the small number of options, this tends to be zero. | If the source data codes ethnicity in an OMOP supported vocabulary, store the concept\_id here. | integer | No | No | Yes | CONCEPT |  |

### observation\_period

**Table Description**

This table contains records which define spans of time during which two conditions are expected to hold: (i) Clinical Events that happened to the Person are recorded in the Event tables, and (ii) absence of records indicate such Events did not occur during this span of time.

**User Guide**

For each Person, one or more OBSERVATION\_PERIOD records may be present, but they will not overlap or be back to back to each other. Events may exist outside all of the time spans of the OBSERVATION\_PERIOD records for a patient, however, absence of an Event outside these time spans cannot be construed as evidence of absence of an Event. Incidence or prevalence rates should only be calculated for the time of active OBSERVATION\_PERIOD records. When constructing cohorts, outside Events can be used for inclusion criteria definition, but without any guarantee for the performance of these criteria. Also, OBSERVATION\_PERIOD records can be as short as a single day, greatly disturbing the denominator of any rate calculation as part of cohort characterizations. To avoid that, apply minimal observation time as a requirement for any cohort definition.

**ETL Conventions**

Each Person needs to have at least one OBSERVATION\_PERIOD record, which should represent time intervals with a high capture rate of Clinical Events. Some source data have very similar concepts, such as enrollment periods in insurance claims data. In other source data such as most EHR systems these time spans need to be inferred under a set of assumptions. It is the discretion of the ETL developer to define these assumptions. In many ETL solutions the start date of the first occurrence or the first high quality occurrence of a Clinical Event (Condition, Drug, Procedure, Device, Measurement, Visit) is defined as the start of the OBSERVATION\_PERIOD record, and the end date of the last occurrence of last high quality occurrence of a Clinical Event, or the end of the database period becomes the end of the OBSERVATOIN\_PERIOD for each Person. If a Person only has a single Clinical Event the OBSERVATION\_PERIOD record can be as short as one day. Depending on these definitions it is possible that Clinical Events fall outside the time spans defined by OBSERVATION\_PERIOD records. Family history or history of Clinical Events generally are not used to generate OBSERVATION\_PERIOD records around the time they are referring to. Any two overlapping or adjacent OBSERVATION\_PERIOD records have to be merged into one.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **observation\_period\_id** | A Person can have multiple discrete Observation Periods which are identified by the Observation\_Period\_Id. | Assign a unique observation\_period\_id to each discrete Observation Period for a Person. | integer | Yes | Yes | No |  |  |
| **person\_id** | The Person ID of the PERSON record for which the Observation Period is recorded. |  | integer | Yes | No | Yes | PERSON |  |
| **observation\_period\_start\_date** | Use this date to determine the start date of the Observation Period. | It is often the case that the idea of Observation Periods does not exist in source data. In those cases, the observation\_period\_start\_date can be inferred as the earliest Event date available for the Person. In insurance claim data, the Observation Period can be considered as the time period the Person is enrolled with a payer. If a Person switches plans but stays with the same payer, and therefore capturing of data continues, that change would be captured in [PAYER\_PLAN\_PERIOD](https://ohdsi.github.io/CommonDataModel/cdm531.html#payer_plan_period). | date | Yes | No | No |  |  |
| **observation\_period\_end\_date** | Use this date to determine the end date of the period for which we can assume that all events for a Person are recorded. | It is often the case that the idea of Observation Periods does not exist in source data. In those cases, the observation\_period\_end\_date can be inferred as the last Event date available for the Person. In insurance claim data, the Observation Period can be considered as the time period the Person is enrolled with a payer. | date | Yes | No | No |  |  |
| **period\_type\_concept\_id** | This field can be used to determine the provenance of the Observation Period as in whether the period was determined from an insurance enrollment file, EHR healthcare encounters, or other sources. | Choose the observation\_period\_type\_concept\_id that best represents how the period was determined. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |

### visit\_occurrence

**Table Description**

This table contains Events where Persons engage with the healthcare system for a duration of time. They are often also called “Encounters”. Visits are defined by a configuration of circumstances under which they occur, such as (i) whether the patient comes to a healthcare institution, the other way around, or the interaction is remote, (ii) whether and what kind of trained medical staff is delivering the service during the Visit, and (iii) whether the Visit is transient or for a longer period involving a stay in bed.

**User Guide**

The configuration defining the Visit are described by Concepts in the Visit Domain, which form a hierarchical structure, but rolling up to generally familiar Visits adopted in most healthcare systems worldwide:

* [Inpatient Visit](https://athena.ohdsi.org/search-terms/terms/9201): Person visiting hospital, at a Care Site, in bed, for duration of more than one day, with physicians and other Providers permanently available to deliver service around the clock
* [Emergency Room Visit](https://athena.ohdsi.org/search-terms/terms/9203): Person visiting dedicated healthcare institution for treating emergencies, at a Care Site, within one day, with physicians and Providers permanently available to deliver service around the clock
* [Emergency Room and Inpatient Visit](https://athena.ohdsi.org/search-terms/terms/262): Person visiting ER followed by a subsequent Inpatient Visit, where Emergency department is part of hospital, and transition from the ER to other hospital departments is undefined
* [Non-hospital institution Visit](https://athena.ohdsi.org/search-terms/terms/42898160): Person visiting dedicated institution for reasons of poor health, at a Care Site, long-term or permanently, with no physician but possibly other Providers permanently available to deliver service around the clock
* [Outpatient Visit](https://athena.ohdsi.org/search-terms/terms/9202): Person visiting dedicated ambulatory healthcare institution, at a Care Site, within one day, without bed, with physicians or medical Providers delivering service during Visit
* [Home Visit](https://athena.ohdsi.org/search-terms/terms/581476): Provider visiting Person, without a Care Site, within one day, delivering service
* [Telehealth Visit](https://athena.ohdsi.org/search-terms/terms/5083): Patient engages with Provider through communication media
* [Pharmacy Visit](https://athena.ohdsi.org/search-terms/terms/581458): Person visiting pharmacy for dispensing of Drug, at a Care Site, within one day
* [Laboratory Visit](https://athena.ohdsi.org/search-terms/terms/32036): Patient visiting dedicated institution, at a Care Site, within one day, for the purpose of a Measurement.
* [Ambulance Visit](https://athena.ohdsi.org/search-terms/terms/581478): Person using transportation service for the purpose of initiating one of the other Visits, without a Care Site, within one day, potentially with Providers accompanying the Visit and delivering service
* [Case Management Visit](https://athena.ohdsi.org/search-terms/terms/38004193): Person interacting with healthcare system, without a Care Site, within a day, with no Providers involved, for administrative purposes

The Visit duration, or ‘length of stay’, is defined as VISIT\_END\_DATE - VISIT\_START\_DATE. For all Visits this is <1 day, except Inpatient Visits and Non-hospital institution Visits. The CDM also contains the VISIT\_DETAIL table where additional information about the Visit is stored, for example, transfers between units during an inpatient Visit.

**ETL Conventions**

Visits can be derived easily if the source data contain coding systems for Place of Service or Procedures, like CPT codes for well visits. In those cases, the codes can be looked up and mapped to a Standard Visit Concept. Otherwise, Visit Concepts have to be identified in the ETL process. This table will contain concepts in the Visit domain. These concepts are arranged in a hierarchical structure to facilitate cohort definitions by rolling up to generally familiar Visits adopted in most healthcare systems worldwide. Visits can be adjacent to each other, i.e. the end date of one can be identical with the start date of the other. As a consequence, more than one-day Visits or their descendants can be recorded for the same day. Multi-day visits must not overlap, i.e. share days other than start and end days. It is often the case that some logic should be written for how to define visits and how to assign Visit\_Concept\_Id. For example, in US claims outpatient visits that appear to occur within the time period of an inpatient visit can be rolled into one with the same Visit\_Occurrence\_Id. In EHR data inpatient visits that are within one day of each other may be strung together to create one visit. It will all depend on the source data and how encounter records should be translated to visit occurrences. Providers can be associated with a Visit through the PROVIDER\_ID field, or indirectly through PROCEDURE\_OCCURRENCE records linked both to the VISIT and PROVIDER tables.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **visit\_occurrence\_id** | Use this to identify unique interactions between a person and the health care system. This identifier links across the other CDM event tables to associate events with a visit. | This should be populated by creating a unique identifier for each unique interaction between a person and the healthcare system where the person receives a medical good or service over a span of time. | integer | Yes | Yes | No |  |  |
| **person\_id** |  |  | integer | Yes | No | Yes | PERSON |  |
| **visit\_concept\_id** | This field contains a concept id representing the kind of visit, like inpatient or outpatient. All concepts in this field should be standard and belong to the Visit domain. | Populate this field based on the kind of visit that took place for the person. For example this could be “Inpatient Visit”, “Outpatient Visit”, “Ambulatory Visit”, etc. This table will contain standard concepts in the Visit domain. These concepts are arranged in a hierarchical structure to facilitate cohort definitions by rolling up to generally familiar Visits adopted in most healthcare systems worldwide. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Visit&standardConcept=Standard&page=1&pageSize=15&query=). | integer | Yes | No | Yes | CONCEPT | Visit |
| **visit\_start\_date** | For inpatient visits, the start date is typically the admission date. For outpatient visits the start date and end date will be the same. | When populating VISIT\_START\_DATE, you should think about the patient experience to make decisions on how to define visits. In the case of an inpatient visit this should be the date the patient was admitted to the hospital or institution. In all other cases this should be the date of the patient-provider interaction. If this information is not available the record should be dropped. | date | Yes | No | No |  |  |
| **visit\_start\_datetime** |  | If no time is given for the start date of a visit, set it to midnight (00:00:0000). | datetime | No | No | No |  |  |
| **visit\_end\_date** | For inpatient visits the end date is typically the discharge date. If a Person is still an inpatient in the hospital at the time of the data extract and does not have a visit\_end\_date, then set the visit\_end\_date to the date of the data pull. | Visit end dates are mandatory. If end dates are not provided in the source there are three ways in which to derive them: - Outpatient Visit: visit\_end\_datetime = visit\_start\_datetime - Emergency Room Visit: visit\_end\_datetime = visit\_start\_datetime - Inpatient Visit: Usually there is information about discharge. If not, you should be able to derive the end date from the sudden decline of activity or from the absence of inpatient procedures/drugs. - Non-hospital institution Visits: Particularly for claims data, if end dates are not provided assume the visit is for the duration of month that it occurs. For Inpatient Visits ongoing at the date of ETL, put date of processing the data into visit\_end\_datetime and visit\_type\_concept\_id with 32220 “Still patient” to identify the visit as incomplete. - All other Visits: visit\_end\_datetime = visit\_start\_datetime. If this is a one-day visit the end date should match the start date. | date | Yes | No | No |  |  |
| **visit\_end\_datetime** | If a Person is still an inpatient in the hospital at the time of the data extract and does not have a visit\_end\_datetime, then set the visit\_end\_datetime to the datetime of the data pull. | If no time is given for the end date of a visit, set it to midnight (00:00:0000). | datetime | No | No | No |  |  |
| **visit\_type\_concept\_id** | Use this field to understand the provenance of the visit record, or where the record comes from. | Populate this field based on the provenance of the visit record, as in whether it came from an EHR record or billing claim. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | Integer | Yes | No | Yes | CONCEPT | Type Concept |
| **provider\_id** | There will only be one provider per visit record and the ETL document should clearly state how they were chosen (attending, admitting, etc.). If there are multiple providers associated with a visit in the source, this can be reflected in the event tables (CONDITION\_OCCURRENCE, PROCEDURE\_OCCURRENCE, etc.) or in the VISIT\_DETAIL table. | If there are multiple providers associated with a visit, you will need to choose which one to put here. The additional providers can be stored in the [VISIT\_DETAIL](https://ohdsi.github.io/CommonDataModel/cdm531.html#visit_detail) table. | integer | No | No | Yes | PROVIDER |  |
| **care\_site\_id** | This field provides information about the Care Site where the Visit took place. | There should only be one Care Site associated with a Visit. | integer | No | No | Yes | CARE\_SITE |  |
| **visit\_source\_value** | This field houses the verbatim value from the source data representing the kind of visit that took place (inpatient, outpatient, emergency, etc.) | If there is information about the kind of visit in the source data that value should be stored here. If a visit is an amalgamation of visits from the source then use a hierarchy to choose the visit source value, such as IP -> ER-> OP. This should line up with the logic chosen to determine how visits are created. | varchar(50) | No | No | No |  |  |
| **visit\_source\_concept\_id** |  | If the visit source value is coded in the source data using an OMOP supported vocabulary put the concept id representing the source value here. | integer | No | No | Yes | CONCEPT |  |
| **admitted\_from\_concept\_id** | Use this field to determine where the patient was admitted from. This concept is part of the visit domain and can indicate if a patient was admitted to the hospital from a long-term care facility, for example. | If available, map the admitted\_from\_source\_value to a standard concept in the visit domain. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Visit&standardConcept=Standard&page=1&pageSize=15&query=). If a person was admitted from home or was self-referred, set this to 0. | integer | No | No | Yes | CONCEPT | Visit |
| **admitted\_from\_source\_value** |  | This information may be called something different in the source data but the field is meant to contain a value indicating where a person was admitted from. Typically this applies only to visits that have a length of stay, like inpatient visits or long-term care visits. | varchar(50) | No | No | No |  |  |
| **discharged\_to\_concept\_id** | Use this field to determine where the patient was discharged to after a visit. This concept is part of the visit domain and can indicate if a patient was transferred to another hospital or sent to a long-term care facility, for example. It is assumed that a person is discharged to home therefore there is not a standard concept id for “home”. Use concept id = 0 when a person is discharged to home. | If available, map the discharged\_to\_source\_value to a standard concept in the visit domain. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Visit&standardConcept=Standard&page=1&pageSize=15&query=). | integer | No | No | Yes | CONCEPT | Visit |
| **discharged\_to\_source\_value** |  | This information may be called something different in the source data but the field is meant to contain a value indicating where a person was discharged to after a visit, as in they went home or were moved to long-term care. Typically this applies only to visits that have a length of stay of a day or more. | varchar(50) | No | No | No |  |  |
| **preceding\_visit\_occurrence\_id** | Use this field to find the visit that occurred for the person prior to the given visit. There could be a few days or a few years in between. | This field can be used to link a visit immediately preceding the current visit. Note this is not symmetrical, and there is no such thing as a “following\_visit\_id”. | integer | No | No | Yes | VISIT\_OCCURRENCE |  |

### visit\_detail

**Table Description**

The VISIT\_DETAIL table is an optional table used to represents details of each record in the parent VISIT\_OCCURRENCE table. A good example of this would be the movement between units in a hospital during an inpatient stay or claim lines associated with a one insurance claim. For every record in the VISIT\_OCCURRENCE table there may be 0 or more records in the VISIT\_DETAIL table with a 1:n relationship where n may be 0. The VISIT\_DETAIL table is structurally very similar to VISIT\_OCCURRENCE table and belongs to the visit domain.

**User Guide**

The configuration defining the Visit Detail is described by Concepts in the Visit Domain, which form a hierarchical structure. The Visit Detail record will have an associated to the Visit Occurrence record in two ways:  
1. The Visit Detail record will have the VISIT\_OCCURRENCE\_ID it is associated to 2. The VISIT\_DETAIL\_CONCEPT\_ID will be a descendant of the VISIT\_CONCEPT\_ID for the Visit.

**ETL Conventions**

It is not mandatory that the VISIT\_DETAIL table be filled in, but if you find that the logic to create VISIT\_OCCURRENCE records includes the roll-up of multiple smaller records to create one picture of a Visit then it is a good idea to use VISIT\_DETAIL. In EHR data, for example, a Person may be in the hospital but instead of one over-arching Visit their encounters are recorded as times they interacted with a health care provider. A Person in the hospital interacts with multiple providers multiple times a day so the encounters must be strung together using some heuristic (defined by the ETL) to identify the entire Visit. In this case the encounters would be considered Visit Details and the entire Visit would be the Visit Occurrence. In this example it is also possible to use the Vocabulary to distinguish Visit Details from a Visit Occurrence by setting the VISIT\_CONCEPT\_ID to [9201](https://athena.ohdsi.org/search-terms/terms/9201) and the VISIT\_DETAIL\_CONCEPT\_IDs either to 9201 or its children to indicate where the patient was in the hospital at the time of care.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **visit\_detail\_id** | Use this to identify unique interactions between a person and the health care system. This identifier links across the other CDM event tables to associate events with a visit detail. | This should be populated by creating a unique identifier for each unique interaction between a person and the healthcare system where the person receives a medical good or service over a span of time. | integer | Yes | Yes | No |  |  |
| **person\_id** |  |  | integer | Yes | No | Yes | PERSON |  |
| **visit\_detail\_concept\_id** | This field contains a concept id representing the kind of visit detail, like inpatient or outpatient. All concepts in this field should be standard and belong to the Visit domain. | Populate this field based on the kind of visit that took place for the person. For example this could be “Inpatient Visit”, “Outpatient Visit”, “Ambulatory Visit”, etc. This table will contain standard concepts in the Visit domain. These concepts are arranged in a hierarchical structure to facilitate cohort definitions by rolling up to generally familiar Visits adopted in most healthcare systems worldwide. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Visit&standardConcept=Standard&page=1&pageSize=15&query=). | integer | Yes | No | Yes | CONCEPT | Visit |
| **visit\_detail\_start\_date** | This is the date of the start of the encounter. This may or may not be equal to the date of the Visit the Visit Detail is associated with. | When populating VISIT\_DETAIL\_START\_DATE, you should think about the patient experience to make decisions on how to define visits. Most likely this should be the date of the patient-provider interaction. | date | Yes | No | No |  |  |
| **visit\_detail\_start\_datetime** |  | If no time is given for the start date of a visit, set it to midnight (00:00:0000). | datetime | No | No | No |  |  |
| **visit\_detail\_end\_date** | This the end date of the patient-provider interaction. If a Person is still an inpatient in the hospital at the time of the data extract and does not have a visit\_end\_date, then set the visit\_end\_date to the date of the data pull. | Visit Detail end dates are mandatory. If end dates are not provided in the source there are three ways in which to derive them:  - Outpatient Visit Detail: visit\_detail\_end\_datetime = visit\_detail\_start\_datetime - Emergency Room Visit Detail: visit\_detail\_end\_datetime = visit\_detail\_start\_datetime - Inpatient Visit Detail: Usually there is information about discharge. If not, you should be able to derive the end date from the sudden decline of activity or from the absence of inpatient procedures/drugs. - Non-hospital institution Visit Details: Particularly for claims data, if end dates are not provided assume the visit is for the duration of month that it occurs.  For Inpatient Visit Details ongoing at the date of ETL, put date of processing the data into visit\_detai\_end\_datetime and visit\_detail\_type\_concept\_id with 32220 “Still patient” to identify the visit as incomplete. All other Visits Details: visit\_detail\_end\_datetime = visit\_detail\_start\_datetime. | date | Yes | No | No |  |  |
| **visit\_detail\_end\_datetime** | If a Person is still an inpatient in the hospital at the time of the data extract and does not have a visit\_end\_datetime, then set the visit\_end\_datetime to the datetime of the data pull. | If no time is given for the end date of a visit, set it to midnight (00:00:0000). | datetime | No | No | No |  |  |
| **visit\_detail\_type\_concept\_id** | Use this field to understand the provenance of the visit detail record, or where the record comes from. | Populate this field based on the provenance of the visit detail record, as in whether it came from an EHR record or billing claim. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |
| **provider\_id** | There will only be one provider per **visit** record and the ETL document should clearly state how they were chosen (attending, admitting, etc.). This is a typical reason for leveraging the VISIT\_DETAIL table as even though each VISIT\_DETAIL record can only have one provider, there is no limit to the number of VISIT\_DETAIL records that can be associated to a VISIT\_OCCURRENCE record. | The additional providers associated to a Visit can be stored in this table where each VISIT\_DETAIL record represents a different provider. | integer | No | No | Yes | PROVIDER |  |
| **care\_site\_id** | This field provides information about the Care Site where the Visit Detail took place. | There should only be one Care Site associated with a Visit Detail. | integer | No | No | Yes | CARE\_SITE |  |
| **visit\_detail\_source\_value** | This field houses the verbatim value from the source data representing the kind of visit detail that took place (inpatient, outpatient, emergency, etc.) | If there is information about the kind of visit detail in the source data that value should be stored here. If a visit is an amalgamation of visits from the source then use a hierarchy to choose the VISIT\_DETAIL\_SOURCE\_VALUE, such as IP -> ER-> OP. This should line up with the logic chosen to determine how visits are created. | varchar(50) | No | No | No |  |  |
| **visit\_detail\_source\_concept\_id** |  | If the VISIT\_DETAIL\_SOURCE\_VALUE is coded in the source data using an OMOP supported vocabulary put the concept id representing the source value here. | Integer | No | No | Yes | CONCEPT |  |
| **admitted\_from\_concept\_id** | Use this field to determine where the patient was admitted from. This concept is part of the visit domain and can indicate if a patient was admitted to the hospital from a long-term care facility, for example. | If available, map the admitted\_from\_source\_value to a standard concept in the visit domain. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Visit&standardConcept=Standard&page=1&pageSize=15&query=). If a person was admitted from home or was self-referred, set this to 0. | Integer | No | No | Yes | CONCEPT | Visit |
| **admitted\_from\_source\_value** |  | This information may be called something different in the source data but the field is meant to contain a value indicating where a person was admitted from. Typically this applies only to visits that have a length of stay, like inpatient visits or long-term care visits. | varchar(50) | No | No | No |  |  |
| **discharged\_to\_source\_value** |  | This information may be called something different in the source data but the field is meant to contain a value indicating where a person was discharged to after a visit, as in they went home or were moved to long-term care. Typically this applies only to visits that have a length of stay of a day or more. | varchar(50) | No | No | No |  |  |
| **discharged\_to\_concept\_id** | Use this field to determine where the patient was discharged to after a visit. This concept is part of the visit domain and can indicate if a patient was transferred to another hospital or sent to a long-term care facility, for example. It is assumed that a person is discharged to home therefore there is not a standard concept id for “home”. Use concept id = 0 when a person is discharged to home. | If available, map the DISCHARGE\_TO\_SOURCE\_VALUE to a Standard Concept in the Visit domain. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Visit&standardConcept=Standard&page=1&pageSize=15&query=). | integer | No | No | Yes | CONCEPT | Visit |
| **preceding\_visit\_detail\_id** | Use this field to find the visit detail that occurred for the person prior to the given visit detail record. There could be a few days or a few years in between. | The PRECEDING\_VISIT\_DETAIL\_ID can be used to link a visit immediately preceding the current Visit Detail. Note this is not symmetrical, and there is no such thing as a “following\_visit\_id”. | integer | No | No | Yes | VISIT\_DETAIL |  |
| **parent\_visit\_detail\_id** | Use this field to find the visit detail that subsumes the given visit detail record. This is used in the case that a visit detail record needs to be nested beyond the VISIT\_OCCURRENCE/VISIT\_DETAIL relationship. | If there are multiple nested levels to how Visits are represented in the source, the VISIT\_DETAIL\_PARENT\_ID can be used to record this relationship. | integer | No | No | Yes | VISIT\_DETAIL |  |
| **visit\_occurrence\_id** | Use this field to link the VISIT\_DETAIL record to its VISIT\_OCCURRENCE. | Put the VISIT\_OCCURRENCE\_ID that subsumes the VISIT\_DETAIL record here. | integer | Yes | No | Yes | VISIT\_OCCURRENCE |  |

### condition\_occurrence

**Table Description**

This table contains records of Events of a Person suggesting the presence of a disease or medical condition stated as a diagnosis, a sign, or a symptom, which is either observed by a Provider or reported by the patient.

**User Guide**

Conditions are defined by Concepts from the Condition domain, which form a complex hierarchy. As a result, the same Person with the same disease may have multiple Condition records, which belong to the same hierarchical family. Most Condition records are mapped from diagnostic codes, but recorded signs, symptoms and summary descriptions also contribute to this table. Rule out diagnoses should not be recorded in this table, but in reality their negating nature is not always captured in the source data, and other precautions must be taken when when identifying Persons who should suffer from the recorded Condition. Record all conditions as they exist in the source data. Any decisions about diagnosis/phenotype definitions would be done through cohort specifications. These cohorts can be housed in the [COHORT](https://ohdsi.github.io/CommonDataModel/cdm531.html#payer_plan_period) table. Conditions span a time interval from start to end, but are typically recorded as single snapshot records with no end date. The reason is twofold: (i) At the time of the recording the duration is not known and later not recorded, and (ii) the Persons typically cease interacting with the healthcare system when they feel better, which leads to incomplete capture of resolved Conditions. The [CONDITION\_ERA](https://ohdsi.github.io/CommonDataModel/cdm531.html#condition_era) table addresses this issue. Family history and past diagnoses (‘history of’) are not recorded in this table. Instead, they are listed in the [OBSERVATION](https://ohdsi.github.io/CommonDataModel/cdm531.html#observation) table. Codes written in the process of establishing the diagnosis, such as ‘question of’ of and ‘rule out’, should not represented here. Instead, they should be recorded in the [OBSERVATION](https://ohdsi.github.io/CommonDataModel/cdm531.html#observation) table, if they are used for analyses. However, this information is not always available.

**ETL Conventions**

Source codes and source text fields mapped to Standard Concepts of the Condition Domain have to be recorded here.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **condition\_occurrence\_id** | The unique key given to a condition record for a person. Refer to the ETL for how duplicate conditions during the same visit were handled. | Each instance of a condition present in the source data should be assigned this unique key. In some cases, a person can have multiple records of the same condition within the same visit. It is valid to keep these duplicates and assign them individual, unique, CONDITION\_OCCURRENCE\_IDs, though it is up to the ETL how they should be handled. | integer | Yes | Yes | No |  |  |
| **person\_id** | The PERSON\_ID of the PERSON for whom the condition is recorded. |  | integer | Yes | No | Yes | PERSON |  |
| **condition\_concept\_id** | The CONDITION\_CONCEPT\_ID field is recommended for primary use in analyses, and must be used for network studies. This is the standard concept mapped from the source value which represents a condition | The CONCEPT\_ID that the CONDITION\_SOURCE\_VALUE maps to. Only records whose source values map to concepts with a domain of “Condition” should go in this table. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Condition&standardConcept=Standard&page=1&pageSize=15&query=). | integer | Yes | No | Yes | CONCEPT | Condition |
| **condition\_start\_date** | Use this date to determine the start date of the condition | Most often data sources do not have the idea of a start date for a condition. Rather, if a source only has one date associated with a condition record it is acceptable to use that date for both the CONDITION\_START\_DATE and the CONDITION\_END\_DATE. | date | Yes | No | No |  |  |
| **condition\_start\_datetime** |  | If a source does not specify datetime the convention is to set the time to midnight (00:00:0000) | datetime | No | No | No |  |  |
| **condition\_end\_date** | Use this date to determine the end date of the condition | Most often data sources do not have the idea of a start date for a condition. Rather, if a source only has one date associated with a condition record it is acceptable to use that date for both the CONDITION\_START\_DATE and the CONDITION\_END\_DATE. | date | No | No | No |  |  |
| **condition\_end\_datetime** |  | If a source does not specify datetime the convention is to set the time to midnight (00:00:0000) | datetime | No | No | No |  |  |
| **condition\_type\_concept\_id** | This field can be used to determine the provenance of the Condition record, as in whether the condition was from an EHR system, insurance claim, registry, or other sources. | Choose the CONDITION\_TYPE\_CONCEPT\_ID that best represents the provenance of the record. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |
| **condition\_status\_concept\_id** | This concept represents the point during the visit the diagnosis was given (admitting diagnosis, final diagnosis), whether the diagnosis was determined due to laboratory findings, if the diagnosis was exclusionary, or if it was a preliminary diagnosis, among others. | Choose the Concept in the Condition Status domain that best represents the point during the visit when the diagnosis was given. These can include admitting diagnosis, principal diagnosis, and secondary diagnosis. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Condition+Status&standardConcept=Standard&page=1&pageSize=15&query=). | integer | No | No | Yes | CONCEPT | Condition Status |
| **stop\_reason** | The Stop Reason indicates why a Condition is no longer valid with respect to the purpose within the source data. Note that a Stop Reason does not necessarily imply that the condition is no longer occurring. | This information is often not populated in source data and it is a valid etl choice to leave it blank if the information does not exist. | varchar(20) | No | No | No |  |  |
| **provider\_id** | The provider associated with condition record, e.g. the provider who made the diagnosis or the provider who recorded the symptom. | The ETL may need to make a choice as to which PROVIDER\_ID to put here. Based on what is available this may or may not be different than the provider associated with the overall VISIT\_OCCURRENCE record, for example the admitting vs attending physician on an EHR record. | integer | No | No | Yes | PROVIDER |  |
| **visit\_occurrence\_id** | The visit during which the condition occurred. | Depending on the structure of the source data, this may have to be determined based on dates. If a CONDITION\_START\_DATE occurs within the start and end date of a Visit it is a valid ETL choice to choose the VISIT\_OCCURRENCE\_ID from the Visit that subsumes it, even if not explicitly stated in the data. While not required, an attempt should be made to locate the VISIT\_OCCURRENCE\_ID of the CONDITION\_OCCURRENCE record. | integer | No | No | Yes | VISIT\_OCCURRENCE |  |
| **visit\_detail\_id** | The VISIT\_DETAIL record during which the condition occurred. For example, if the person was in the ICU at the time of the diagnosis the VISIT\_OCCURRENCE record would reflect the overall hospital stay and the VISIT\_DETAIL record would reflect the ICU stay during the hospital visit. | Same rules apply as for the VISIT\_OCCURRENCE\_ID. | integer | No | No | Yes | VISIT\_DETAIL |  |
| **condition\_source\_value** | This field houses the verbatim value from the source data representing the condition that occurred. For example, this could be an ICD10 or Read code. | This code is mapped to a Standard Condition Concept in the Standardized Vocabularies and the original code is stored here for reference. | varchar(50) | No | No | No |  |  |
| **condition\_source\_concept\_id** | This is the concept representing the condition source value and may not necessarily be standard. This field is discouraged from use in analysis because it is not required to contain Standard Concepts that are used across the OHDSI community, and should only be used when Standard Concepts do not adequately represent the source detail for the Condition necessary for a given analytic use case. Consider using CONDITION\_CONCEPT\_ID instead to enable standardized analytics that can be consistent across the network. | If the CONDITION\_SOURCE\_VALUE is coded in the source data using an OMOP supported vocabulary put the concept id representing the source value here. | integer | No | No | Yes | CONCEPT |  |
| **condition\_status\_source\_value** | This field houses the verbatim value from the source data representing the condition status. | This information may be called something different in the source data but the field is meant to contain a value indicating when and how a diagnosis was given to a patient. This source value is mapped to a standard concept which is stored in the CONDITION\_STATUS\_CONCEPT\_ID field. | varchar(50) | No | No | No |  |  |

### drug\_exposure

**Table Description**

This table captures records about the exposure to a Drug ingested or otherwise introduced into the body. A Drug is a biochemical substance formulated in such a way that when administered to a Person it will exert a certain biochemical effect on the metabolism. Drugs include prescription and over-the-counter medicines, vaccines, and large-molecule biologic therapies. Radiological devices ingested or applied locally do not count as Drugs.

**User Guide**

The purpose of records in this table is to indicate an exposure to a certain drug as best as possible. In this context a drug is defined as an active ingredient. Drug Exposures are defined by Concepts from the Drug domain, which form a complex hierarchy. As a result, one DRUG\_SOURCE\_CONCEPT\_ID may map to multiple standard concept ids if it is a combination product. Records in this table represent prescriptions written, prescriptions dispensed, and drugs administered by a provider to name a few. The DRUG\_TYPE\_CONCEPT\_ID can be used to find and filter on these types. This table includes additional information about the drug products, the quantity given, and route of administration.

**ETL Conventions**

Information about quantity and dose is provided in a variety of different ways and it is important for the ETL to provide as much information as possible from the data. Depending on the provenance of the data fields may be captured differently i.e. quantity for drugs administered may have a separate meaning from quantity for prescriptions dispensed. If a patient has multiple records on the same day for the same drug or procedures the ETL should not de-dupe them unless there is probable reason to believe the item is a true data duplicate. Take note on how to handle refills for prescriptions written.  
  
For detailed conventions on how to populate this table, please refer to the [THEMIS repository](https://ohdsi.github.io/Themis/drug_exposure.html).

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **drug\_exposure\_id** | The unique key given to records of drug dispensings or administrations for a person. Refer to the ETL for how duplicate drugs during the same visit were handled. | Each instance of a drug dispensing or administration present in the source data should be assigned this unique key. In some cases, a person can have multiple records of the same drug within the same visit. It is valid to keep these duplicates and assign them individual, unique, DRUG\_EXPOSURE\_IDs, though it is up to the ETL how they should be handled. | integer | Yes | Yes | No |  |  |
| **person\_id** | The PERSON\_ID of the PERSON for whom the drug dispensing or administration is recorded. This may be a system generated code. |  | integer | Yes | No | Yes | PERSON |  |
| **drug\_concept\_id** | The DRUG\_CONCEPT\_ID field is recommended for primary use in analyses, and must be used for network studies. This is the standard concept mapped from the source concept id which represents a drug product or molecule otherwise introduced to the body. The drug concepts can have a varying degree of information about drug strength and dose. This information is relevant in the context of quantity and administration information in the subsequent fields plus strength information from the DRUG\_STRENGTH table, provided as part of the standard vocabulary download. | The CONCEPT\_ID that the DRUG\_SOURCE\_VALUE maps to. The concept id should be derived either from mapping from the source concept id or by picking the drug concept representing the most amount of detail you have. Records whose source values map to standard concepts with a domain of Drug should go in this table. 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 most detailed content of information is preferred during the mapping process. These are indicated in the CONCEPT\_CLASS\_ID field of the Concept and are recorded in the following order of precedence: Marketed Product, Branded Pack, Clinical Pack, Branded Drug, Clinical Drug, Branded Drug Component, Clinical Drug Component, Branded Drug Form, Clinical Drug Form, and only if no other information is available Ingredient. Note: If only the drug class is known, the DRUG\_CONCEPT\_ID field should contain 0. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Drug&standardConcept=Standard&page=1&pageSize=15&query=). | integer | Yes | No | Yes | CONCEPT | Drug |
| **drug\_exposure\_start\_date** | Use this date to determine the start date of the drug record. | Valid entries include a start date of a prescription, the date a prescription was filled, or the date on which a Drug administration was recorded. It is a valid ETL choice to use the date the drug was ordered as the DRUG\_EXPOSURE\_START\_DATE. | date | Yes | No | No |  |  |
| **drug\_exposure\_start\_datetime** |  | This is not required, though it is in v6. If a source does not specify datetime the convention is to set the time to midnight (00:00:0000) | datetime | No | No | No |  |  |
| **drug\_exposure\_end\_date** | The DRUG\_EXPOSURE\_END\_DATE denotes the day the drug exposure ended for the patient. | If this information is not explicitly available in the data, infer the end date using the following methods:  1. Start first with duration or days supply using the calculation drug start date + days supply -1 day. 2. Use quantity divided by daily dose that you may obtain from the sig or a source field (or assumed daily dose of 1) for solid, indivisibile, drug products. If quantity represents ingredient amount, quantity divided by daily dose \* concentration (from drug\_strength) drug concept id tells you the dose form. 3. If it is an administration record, set drug end date equal to drug start date. If the record is a written prescription then set end date to start date + 29. If the record is a mail-order prescription set end date to start date + 89. The end date must be equal to or greater than the start date. Ibuprofen 20mg/mL oral solution concept tells us this is oral solution. Calculate duration as quantity (200 example) \* daily dose (5mL) /concentration (20mg/mL) 200\*5/20 = 50 days. [Examples by dose form](https://ohdsi.github.io/CommonDataModel/drug_dose.html)  For detailed conventions for how to populate this field, please see the [THEMIS repository](https://ohdsi.github.io/Themis/tag_drug_exposure.html). | date | Yes | No | No |  |  |
| **drug\_exposure\_end\_datetime** |  | This is not required, though it is in v6. If a source does not specify datetime the convention is to set the time to midnight (00:00:0000) | datetime | No | No | No |  |  |
| **verbatim\_end\_date** | This is the end date of the drug exposure as it appears in the source data, if it is given | Put the end date or discontinuation date as it appears from the source data or leave blank if unavailable. | date | No | No | No |  |  |
| **drug\_type\_concept\_id** | You can use the TYPE\_CONCEPT\_ID to delineate between prescriptions written vs. prescriptions dispensed vs. medication history vs. patient-reported exposure, etc. | Choose the drug\_type\_concept\_id that best represents the provenance of the record, for example whether it came from a record of a prescription written or physician administered drug. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |
| **stop\_reason** | The reason a person stopped a medication as it is represented in the source. Reasons include regimen completed, changed, removed, etc. This field will be retired in v6.0. | This information is often not populated in source data and it is a valid etl choice to leave it blank if the information does not exist. | varchar(20) | No | No | No |  |  |
| **refills** | This is only filled in when the record is coming from a prescription written this field is meant to represent intended refills at time of the prescription. |  | integer | No | No | No |  |  |
| **quantity** |  | To find the dose form of a drug the RELATIONSHIP table can be used where the relationship\_id is ‘Has dose form’. If liquid, quantity stands for the total amount dispensed or ordered of ingredient in the units given by the drug\_strength table. If the unit from the source data does not align with the unit in the DRUG\_STRENGTH table the quantity should be converted to the correct unit given in DRUG\_STRENGTH. For clinical drugs with fixed dose forms (tablets etc.) the quantity is the number of units/tablets/capsules prescribed or dispensed (can be partial, but then only 1/2 or 1/3, not 0.01). Clinical drugs with divisible dose forms (injections) the quantity is the amount of ingredient the patient got. For example, if the injection is 2mg/mL but the patient got 80mL then quantity is reported as 160. Quantified clinical drugs with divisible dose forms (prefilled syringes), the quantity is the amount of ingredient similar to clinical drugs. Please see [how to calculate drug dose](https://ohdsi.github.io/CommonDataModel/drug_dose.html) for more information. | float | No | No | No |  |  |
| **days\_supply** |  | The number of days of supply of the medication as recorded in the original prescription or dispensing record. Days supply can differ from actual drug duration (i.e. prescribed days supply vs actual exposure).”,“The field should be left empty if the source data does not contain a verbatim days\_supply, and should not be calculated from other fields.  Negative values are not allowed. If the source has negative days supply the record should be dropped as it is unknown if the patient actually took the drug. Several actions are possible: 1) record is not trustworthy and we remove the record entirely. 2) we trust the record and leave days\_supply empty or 3) record needs to be combined with other record (e.g. reversal of prescription). High values (>365 days) should be investigated. If considered an error in the source data (e.g. typo), the value needs to be excluded to prevent creation of unrealistic long eras. | integer | No | No | No |  |  |
| **sig** | This is the verbatim instruction for the drug as written by the provider. | Put the written out instructions for the drug as it is verbatim in the source, if available. | varchar(MAX) | No | No | No |  |  |
| **route\_concept\_id** |  | The standard CONCEPT\_ID that the ROUTE\_SOURCE\_VALUE maps to in the route domain. | integer | No | No | Yes | CONCEPT | Route |
| **lot\_number** |  |  | varchar(50) | No | No | No |  |  |
| **provider\_id** | The Provider associated with drug record, e.g. the provider who wrote the prescription or the provider who administered the drug. | The ETL may need to make a choice as to which PROVIDER\_ID to put here. Based on what is available this may or may not be different than the provider associated with the overall VISIT\_OCCURRENCE record, for example the ordering vs administering physician on an EHR record. | integer | No | No | Yes | PROVIDER |  |
| **visit\_occurrence\_id** | The Visit during which the drug was prescribed, administered or dispensed. | To populate this field drug exposures must be explicitly initiated in the visit. | integer | No | No | Yes | VISIT\_OCCURRENCE |  |
| **visit\_detail\_id** | The VISIT\_DETAIL record during which the drug exposure occurred. For example, if the person was in the ICU at the time of the drug administration the VISIT\_OCCURRENCE record would reflect the overall hospital stay and the VISIT\_DETAIL record would reflect the ICU stay during the hospital visit. | Same rules apply as for the VISIT\_OCCURRENCE\_ID. | integer | No | No | Yes | VISIT\_DETAIL |  |
| **drug\_source\_value** | This field houses the verbatim value from the source data representing the drug exposure that occurred. For example, this could be an NDC or Gemscript code. | This code is mapped to a Standard Drug Concept in the Standardized Vocabularies and the original code is stored here for reference. | varchar(50) | No | No | No |  |  |
| **drug\_source\_concept\_id** | This is the concept representing the drug source value and may not necessarily be standard. This field is discouraged from use in analysis because it is not required to contain Standard Concepts that are used across the OHDSI community, and should only be used when Standard Concepts do not adequately represent the source detail for the Drug necessary for a given analytic use case. Consider using DRUG\_CONCEPT\_ID instead to enable standardized analytics that can be consistent across the network. | If the DRUG\_SOURCE\_VALUE is coded in the source data using an OMOP supported vocabulary put the concept id representing the source value here. | integer | No | No | Yes | CONCEPT |  |
| **route\_source\_value** | This field houses the verbatim value from the source data representing the drug route. | This information may be called something different in the source data but the field is meant to contain a value indicating when and how a drug was given to a patient. This source value is mapped to a standard concept which is stored in the ROUTE\_CONCEPT\_ID field. | varchar(50) | No | No | No |  |  |
| **dose\_unit\_source\_value** | This field houses the verbatim value from the source data representing the dose unit of the drug given. | This information may be called something different in the source data but the field is meant to contain a value indicating the unit of dosage of drug given to the patient. **This is an older column and will be deprecated in an upcoming version.** | varchar(50) | No | No | No |  |  |

### procedure\_occurrence

**Table Description**

This table contains records of activities or processes ordered by, or carried out by, a healthcare provider on the patient with a diagnostic or therapeutic purpose.

**User Guide**

Lab tests are not a procedure, if something is observed with an expected resulting amount and unit then it should be a measurement. Phlebotomy is a procedure but so trivial that it tends to be rarely captured. It can be assumed that there is a phlebotomy procedure associated with many lab tests, therefore it is unnecessary to add them as separate procedures. If the user finds the same procedure over concurrent days, it is assumed those records are part of a procedure lasting more than a day. This logic is in lieu of the procedure\_end\_date, which will be added in a future version of the CDM.

**ETL Conventions**

When dealing with duplicate records, the ETL must determine whether to sum them up into one record or keep them separate. Things to consider are: - Same Procedure - Same PROCEDURE\_DATETIME - Same Visit Occurrence or Visit Detail - Same Provider - Same Modifier for Procedures. Source codes and source text fields mapped to Standard Concepts of the Procedure Domain have to be recorded here.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **procedure\_occurrence\_id** | The unique key given to a procedure record for a person. Refer to the ETL for how duplicate procedures during the same visit were handled. | Each instance of a procedure occurrence in the source data should be assigned this unique key. In some cases, a person can have multiple records of the same procedure within the same visit. It is valid to keep these duplicates and assign them individual, unique, PROCEDURE\_OCCURRENCE\_IDs, though it is up to the ETL how they should be handled. | integer | Yes | Yes | No |  |  |
| **person\_id** | The PERSON\_ID of the PERSON for whom the procedure is recorded. This may be a system generated code. |  | integer | Yes | No | Yes | PERSON |  |
| **procedure\_concept\_id** | The PROCEDURE\_CONCEPT\_ID field is recommended for primary use in analyses, and must be used for network studies. This is the standard concept mapped from the source value which represents a procedure | The CONCEPT\_ID that the PROCEDURE\_SOURCE\_VALUE maps to. Only records whose source values map to standard concepts with a domain of “Procedure” should go in this table. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Procedure&standardConcept=Standard&page=1&pageSize=15&query=). | integer | Yes | No | Yes | CONCEPT | Procedure |
| **procedure\_date** | Use this date to determine the date the procedure started. | This is meant to be the **start date** of the procedure. It will be renamed in a future version to **PROCEDURE\_START\_DATE**. | date | Yes | No | No |  |  |
| **procedure\_datetime** |  | If the procedure has a start time in the native date, use this field to house that information. This will be renamed in a future version to **PROCEDURE\_START\_DATETIME**. | datetime | No | No | No |  |  |
| **procedure\_end\_date** | Use this field to house the date that the procedure ended. | This is meant to be the end date of the procedure. It is not required and for most cases will be the same as the PROCEDURE\_START\_DATE. | date | No | No | No |  |  |
| **procedure\_end\_datetime** | Use this field to house the datetime that the procedure ended. | This is meant to house the end datetime of the procedure and will most often be used in conjunction with the procedure\_start\_datetime to determine the length of the procedure. | datetime | No | No | No |  |  |
| **procedure\_type\_concept\_id** | This field can be used to determine the provenance of the Procedure record, as in whether the procedure was from an EHR system, insurance claim, registry, or other sources. | Choose the PROCEDURE\_TYPE\_CONCEPT\_ID that best represents the provenance of the record, for example whether it came from an EHR record or billing claim. If a procedure is recorded as an EHR encounter, the PROCEDURE\_TYPE\_CONCEPT would be ‘EHR encounter record’. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |
| **modifier\_concept\_id** | The modifiers are intended to give additional information about the procedure but as of now the vocabulary is under review. | It is up to the ETL to choose how to map modifiers if they exist in source data. These concepts are typically distinguished by ‘Modifier’ concept classes (e.g., ‘CPT4 Modifier’ as part of the ‘CPT4’ vocabulary). If there is more than one modifier on a record, one should be chosen that pertains to the procedure rather than provider. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?conceptClass=CPT4+Modifier&conceptClass=HCPCS+Modifier&vocabulary=CPT4&vocabulary=HCPCS&standardConcept=Standard&page=1&pageSize=15&query=). | integer | No | No | Yes | CONCEPT |  |
| **quantity** | If the quantity value is omitted, a single procedure is assumed. | If a Procedure has a quantity of ‘0’ in the source, this should default to ‘1’ in the ETL. If there is a record in the source it can be assumed the exposure occurred at least once | integer | No | No | No |  |  |
| **provider\_id** | The provider associated with the procedure record, e.g. the provider who performed the Procedure. | The ETL may need to make a choice as to which PROVIDER\_ID to put here. Based on what is available this may or may not be different than the provider associated with the overall VISIT\_OCCURRENCE record, for example the admitting vs attending physician on an EHR record. | integer | No | No | Yes | PROVIDER |  |
| **visit\_occurrence\_id** | The visit during which the procedure occurred. | Depending on the structure of the source data, this may have to be determined based on dates. If a PROCEDURE\_DATE occurs within the start and end date of a Visit it is a valid ETL choice to choose the VISIT\_OCCURRENCE\_ID from the Visit that subsumes it, even if not explicitly stated in the data. While not required, an attempt should be made to locate the VISIT\_OCCURRENCE\_ID of the PROCEDURE\_OCCURRENCE record. | integer | No | No | Yes | VISIT\_OCCURRENCE |  |
| **visit\_detail\_id** | The VISIT\_DETAIL record during which the Procedure occurred. For example, if the Person was in the ICU at the time of the Procedure the VISIT\_OCCURRENCE record would reflect the overall hospital stay and the VISIT\_DETAIL record would reflect the ICU stay during the hospital visit. | Same rules apply as for the VISIT\_OCCURRENCE\_ID. | integer | No | No | Yes | VISIT\_DETAIL |  |
| **procedure\_source\_value** | This field houses the verbatim value from the source data representing the procedure that occurred. For example, this could be an CPT4 or OPCS4 code. | Use this value to look up the source concept id and then map the source concept id to a standard concept id. | varchar(50) | No | No | No |  |  |
| **procedure\_source\_concept\_id** | This is the concept representing the procedure source value and may not necessarily be standard. This field is discouraged from use in analysis because it is not required to contain Standard Concepts that are used across the OHDSI community, and should only be used when Standard Concepts do not adequately represent the source detail for the Procedure necessary for a given analytic use case. Consider using PROCEDURE\_CONCEPT\_ID instead to enable standardized analytics that can be consistent across the network. | If the PROCEDURE\_SOURCE\_VALUE is coded in the source data using an OMOP supported vocabulary put the concept id representing the source value here. | integer | No | No | Yes | CONCEPT |  |
| **modifier\_source\_value** | This field houses the verbatim value from the source data representing the modifier code for the procedure that occurred. | The value stored here is mapped to a MODIFIER\_CONCEPT\_ID. | varchar(50) | No | No | No |  |  |

### device\_exposure

**Table Description**

The Device domain captures information about a person’s exposure to a foreign physical object or instrument which is used for diagnostic or therapeutic purposes through a mechanism beyond chemical action. Devices include implantable objects (e.g. pacemakers, stents, artificial joints), medical equipment and supplies (e.g. bandages, crutches, syringes), other instruments used in medical procedures (e.g. sutures, defibrillators) and material used in clinical care (e.g. adhesives, body material, dental material, surgical material).

**User Guide**

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.

**ETL Conventions**

Source codes and source text fields mapped to Standard Concepts of the Device Domain have to be recorded here.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **device\_exposure\_id** | The unique key given to records a person’s exposure to a foreign physical object or instrument. | Each instance of an exposure to a foreign object or device present in the source data should be assigned this unique key. | integer | Yes | Yes | No |  |  |
| **person\_id** |  |  | integer | Yes | No | Yes | PERSON |  |
| **device\_concept\_id** | The DEVICE\_CONCEPT\_ID field is recommended for primary use in analyses, and must be used for network studies. This is the standard concept mapped from the source concept id which represents a foreign object or instrument the person was exposed to. | The CONCEPT\_ID that the DEVICE\_SOURCE\_VALUE maps to. | integer | Yes | No | Yes | CONCEPT | Device |
| **device\_exposure\_start\_date** | Use this date to determine the start date of the device record. | Valid entries include a start date of a procedure to implant a device, the date of a prescription for a device, or the date of device administration. | date | Yes | No | No |  |  |
| **device\_exposure\_start\_datetime** |  | This is not required, though it is in v6. If a source does not specify datetime the convention is to set the time to midnight (00:00:0000) | datetime | No | No | No |  |  |
| **device\_exposure\_end\_date** | The DEVICE\_EXPOSURE\_END\_DATE denotes the day the device exposure ended for the patient, if given. | Put the end date or discontinuation date as it appears from the source data or leave blank if unavailable. | date | No | No | No |  |  |
| **device\_exposure\_end\_datetime** |  | If a source does not specify datetime the convention is to set the time to midnight (00:00:0000) | datetime | No | No | No |  |  |
| **device\_type\_concept\_id** | You can use the TYPE\_CONCEPT\_ID to denote the provenance of the record, as in whether the record is from administrative claims or EHR. | Choose the drug\_type\_concept\_id that best represents the provenance of the record, for example whether it came from a record of a prescription written or physician administered drug. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |
| **unique\_device\_id** | This is the Unique Device Identification (UDI-DI) number for devices regulated by the FDA, if given. | For medical devices that are regulated by the FDA, a Unique Device Identification (UDI) is provided if available in the data source and is recorded in the UNIQUE\_DEVICE\_ID field. | varchar(255) | No | No | No |  |  |
| **production\_id** | This is the Production Identifier (UDI-PI) portion of the Unique Device Identification. |  | varchar(255) | No | No | No |  |  |
| **quantity** |  | If there is a record of device exposure in the source but no quantity value, then set to 1. | integer | No | No | No |  |  |
| **provider\_id** | The Provider associated with device record, e.g. the provider who wrote the prescription or the provider who implanted the device. | The ETL may need to make a choice as to which PROVIDER\_ID to put here. Based on what is available this may or may not be different than the provider associated with the overall VISIT\_OCCURRENCE record. | integer | No | No | Yes | PROVIDER |  |
| **visit\_occurrence\_id** | The Visit during which the device was prescribed or given. | To populate this field device exposures must be explicitly initiated in the visit. | integer | No | No | Yes | VISIT\_OCCURRENCE |  |
| **visit\_detail\_id** | The Visit Detail during which the device was prescribed or given. | To populate this field device exposures must be explicitly initiated in the visit detail record. | integer | No | No | Yes | VISIT\_DETAIL |  |
| **device\_source\_value** | This field houses the verbatim value from the source data representing the device exposure that occurred. For example, this could be an NDC or Gemscript code. | This code is mapped to a Standard Device Concept in the Standardized Vocabularies and the original code is stored here for reference. | varchar(50) | No | No | No |  |  |
| **device\_source\_concept\_id** | This is the concept representing the device source value and may not necessarily be standard. This field is discouraged from use in analysis because it is not required to contain Standard Concepts that are used across the OHDSI community, and should only be used when Standard Concepts do not adequately represent the source detail for the Device necessary for a given analytic use case. Consider using DEVICE\_CONCEPT\_ID instead to enable standardized analytics that can be consistent across the network. | If the DEVICE\_SOURCE\_VALUE is coded in the source data using an OMOP supported vocabulary put the concept id representing the source value here. | integer | No | No | Yes | CONCEPT |  |
| **unit\_concept\_id** | UNIT\_SOURCE\_VALUES should be mapped to a Standard Concept in the Unit domain that best represents the unit as given in the source data. | There is no standardization requirement for units associated with DEVICE\_CONCEPT\_IDs, however, it is the responsibility of the ETL to choose the most plausible unit. If the source unit is NULL (applicable to cases when there’s no numerical value or when it doesn’t require a unit), keep unit\_concept\_id NULL as well. If there’s no mapping of a source unit, populate unit\_concept\_id with 0. | integer | No | No | Yes | CONCEPT | Unit |
| **unit\_source\_value** | This field houses the verbatim value from the source data representing the unit of the Device. For example, blood transfusions are considered devices and can be given in mL quantities. | This code is mapped to a Standard Condition Concept in the Standardized Vocabularies and the original code is stored here for reference. Using the blood transfusion example, blood transfusion is represented by the DEVICE\_CONCEPT\_ID and the unit (mL) would be housed in the UNIT\_SOURCE\_VALUE and mapped to a standard concept in the unit domain. | varchar(50) | No | No | No |  |  |
| **unit\_source\_concept\_id** | This is the concept representing the UNIT\_SOURCE\_VALUE and may not necessarily be standard. This field is discouraged from use in analysis because it is not required to contain Standard Concepts that are used across the OHDSI community, and should only be used when Standard Concepts do not adequately represent the source detail for the Unit necessary for a given analytic use case. Consider using UNIT\_CONCEPT\_ID instead to enable standardized analytics that can be consistent across the network. | If the UNIT\_SOURCE\_VALUE is coded in the source data using an OMOP supported vocabulary put the concept id representing the source value here. | integer | No | No | Yes | CONCEPT |  |

### measurement

**Table Description**

The MEASUREMENT table contains records of Measurements, i.e. structured values (numerical or categorical) obtained through systematic and standardized examination or testing of a Person or Person’s sample. The MEASUREMENT table contains both orders and results of such Measurements as laboratory tests, vital signs, quantitative findings from pathology reports, etc. Measurements are stored as attribute value pairs, with the attribute as the Measurement Concept and the value representing the result. The value can be a Concept (stored in VALUE\_AS\_CONCEPT), or a numerical value (VALUE\_AS\_NUMBER) with a Unit (UNIT\_CONCEPT\_ID). The Procedure for obtaining the sample is housed in the PROCEDURE\_OCCURRENCE table, though it is unnecessary to create a PROCEDURE\_OCCURRENCE record for each measurement if one does not exist in the source data. Measurements differ from Observations in that they require a standardized test or some other activity to generate a quantitative or qualitative result. If there is no result, it is assumed that the lab test was conducted but the result was not captured.

**User Guide**

Measurements are predominately lab tests with a few exceptions, like blood pressure or function tests. Results are given in the form of a value and unit combination. When investigating measurements, look for operator\_concept\_ids (<, >, etc.).

**ETL Conventions**

Only records where the source value maps to a Concept in the measurement domain should be included in this table. Even though each Measurement always has a result, the fields VALUE\_AS\_NUMBER and VALUE\_AS\_CONCEPT\_ID are not mandatory as often the result is not given in the source data. When the result is not known, the Measurement record represents just the fact that the corresponding Measurement was carried out, which in itself is already useful information for some use cases. For some Measurement Concepts, the result is included in the test. For example, ICD10 CONCEPT\_ID [45548980](https://athena.ohdsi.org/search-terms/terms/45548980) ‘Abnormal level of unspecified serum enzyme’ indicates a Measurement and the result (abnormal). In those situations, the CONCEPT\_RELATIONSHIP table in addition to the ‘Maps to’ record contains a second record with the relationship\_id set to ‘Maps to value’. In this example, the ‘Maps to’ relationship directs to [4046263](https://athena.ohdsi.org/search-terms/terms/4046263) ‘Enzyme measurement’ as well as a ‘Maps to value’ record to [4135493](https://athena.ohdsi.org/search-terms/terms/4135493) ‘Abnormal’.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **measurement\_id** | The unique key given to a Measurement record for a Person. Refer to the ETL for how duplicate Measurements during the same Visit were handled. | Each instance of a measurement present in the source data should be assigned this unique key. In some cases, a person can have multiple records of the same measurement within the same visit. It is valid to keep these duplicates and assign them individual, unique, MEASUREMENT\_IDs, though it is up to the ETL how they should be handled. | integer | Yes | Yes | No |  |  |
| **person\_id** | The PERSON\_ID of the Person for whom the Measurement is recorded. This may be a system generated code. |  | integer | Yes | No | Yes | PERSON |  |
| **measurement\_concept\_id** | The MEASUREMENT\_CONCEPT\_ID field is recommended for primary use in analyses, and must be used for network studies. This is the standard concept mapped from the source value which represents a measurement. | The CONCEPT\_ID that the MEASUREMENT\_SOURCE\_VALUE maps to. Only records whose source values map to concepts with a domain of Measurement should go in this table. | integer | Yes | No | Yes | CONCEPT | Measurement |
| **measurement\_date** | Use this date to determine the date of the measurement. | If there are multiple dates in the source data associated with a record such as order\_date, draw\_date, and result\_date, choose the one that is closest to the date the sample was drawn from the patient. | date | Yes | No | No |  |  |
| **measurement\_datetime** |  | This is not required, though it is in v6. If a source does not specify datetime the convention is to set the time to midnight (00:00:0000) | datetime | No | No | No |  |  |
| **measurement\_time** |  | This is present for backwards compatibility and will be deprecated in an upcoming version. | varchar(10) | No | No | No |  |  |
| **measurement\_type\_concept\_id** | This field can be used to determine the provenance of the Measurement record, as in whether the measurement was from an EHR system, insurance claim, registry, or other sources. | Choose the MEASUREMENT\_TYPE\_CONCEPT\_ID that best represents the provenance of the record, for example whether it came from an EHR record or billing claim. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |
| **operator\_concept\_id** | The meaning of Concept [4172703](https://athena.ohdsi.org/search-terms/terms/4172703) for ‘=’ is identical to omission of a OPERATOR\_CONCEPT\_ID value. Since the use of this field is rare, it’s important when devising analyses to not to forget testing for the content of this field for values different from =. | Operators are =, > and these concepts belong to the ‘Meas Value Operator’ domain. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Meas+Value+Operator&standardConcept=Standard&page=1&pageSize=15&query=). Leave it NULL if there’s an exact numeric value given (instead of putting ‘=’) or there’s no numeric value at all. | integer | No | No | Yes | CONCEPT |  |
| **value\_as\_number** | This is the numerical value of the Result of the Measurement, if available. Note that measurements such as blood pressures will be split into their component parts i.e. one record for systolic, one record for diastolic. | [Convention for negative values](https://ohdsi.github.io/Themis/negative_value_as_number.html) | float | No | No | No |  |  |
| **value\_as\_concept\_id** | If the raw data gives a categorial result for measurements those values are captured and mapped to standard concepts in the ‘Meas Value’ domain. | If there is no categorial result in the source data, set VALUE\_AS\_CONCEPT\_ID to NULL, if there is a categorial result in a source data but without mapping, set VALUE\_AS\_CONCEPT\_ID to 0, else map to a CONCEPT\_ID. | integer | No | No | Yes | CONCEPT |  |
| **unit\_concept\_id** | At present, there isn’t a prescribed unit for individual measurements, such as Hemoglobin A1C, meaning it’s not obligatory to express these measurements as a percentage. UNIT\_SOURCE\_VALUES should be linked to a Standard Concept within the Unit domain that most accurately reflects the unit provided in the source data. | If the source data does not include units, set UNIT\_CONCEPT\_ID to NULL. If units are provided but not mapped, set UNIT\_CONCEPT\_ID to 0. Otherwise, map the units to a CONCEPT\_ID. Remember that units are case-sensitive in vocabulary. | integer | No | No | Yes | CONCEPT | Unit |
| **range\_low** | Ranges have the same unit as the VALUE\_AS\_NUMBER. These ranges are provided by the source and should remain NULL if not given. | If reference ranges for upper and lower limit of normal as provided (typically by a laboratory) these are stored in the RANGE\_HIGH and RANGE\_LOW fields. This should be set to NULL if not provided. | float | No | No | No |  |  |
| **range\_high** | Ranges have the same unit as the VALUE\_AS\_NUMBER. These ranges are provided by the source and should remain NULL if not given. | If reference ranges for upper and lower limit of normal as provided (typically by a laboratory) these are stored in the RANGE\_HIGH and RANGE\_LOW fields. This should be set to NULL if not provided. | float | No | No | No |  |  |
| **provider\_id** | The provider associated with measurement record, e.g. the provider who ordered the test or the provider who recorded the result. | The ETL may need to make a choice as to which PROVIDER\_ID to put here. Based on what is available this may or may not be different than the provider associated with the overall VISIT\_OCCURRENCE record. For example the admitting vs attending physician on an EHR record. | integer | No | No | Yes | PROVIDER |  |
| **visit\_occurrence\_id** | The visit during which the Measurement occurred. | Depending on the structure of the source data, this may have to be determined based on dates. If a MEASUREMENT\_DATE occurs within the start and end date of a Visit it is a valid ETL choice to choose the VISIT\_OCCURRENCE\_ID from the visit that subsumes it, even if not explicitly stated in the data. While not required, an attempt should be made to locate the VISIT\_OCCURRENCE\_ID of the measurement record. If a measurement is related to a visit explicitly in the source data, it is possible that the result date of the Measurement falls outside of the bounds of the Visit dates. | integer | No | No | Yes | VISIT\_OCCURRENCE |  |
| **visit\_detail\_id** | The VISIT\_DETAIL record during which the Measurement occurred. For example, if the Person was in the ICU at the time the VISIT\_OCCURRENCE record would reflect the overall hospital stay and the VISIT\_DETAIL record would reflect the ICU stay during the hospital visit. | Same rules apply as for the VISIT\_OCCURRENCE\_ID. | integer | No | No | Yes | VISIT\_DETAIL |  |
| **measurement\_source\_value** | This field contains the exact value from the source data that represents the measurement that occurred. | This value corresponds to a standardized CONCEPT\_ID found in MEASUREMENT\_CONCEPT\_ID and in the ‘Measurement’ domain within the Standardized Vocabularies. The original code is retained here for reference purposes. | varchar(50) | No | No | No |  |  |
| **measurement\_source\_concept\_id** | This is the concept representing the MEASUREMENT\_SOURCE\_VALUE and may not necessarily be standard. This field is discouraged from use in analysis because it is not required to contain Standard Concepts that are used across the OHDSI community, and should only be used when Standard Concepts do not adequately represent the source detail for the Measurement necessary for a given analytic use case. Consider using MEASUREMENT\_CONCEPT\_ID instead to enable standardized analytics that can be consistent across the network. | If the MEASUREMENT\_SOURCE\_VALUE is coded in the source data using a vocabulary supported by OMOP Standardized Vocabularies, insert the CONCEPT\_ID representing the source value here. | integer | No | No | Yes | CONCEPT |  |
| **unit\_source\_value** | This field contains the exact value from the source data that represents the unit of measurement used. | This value corresponds to a standardized CONCEPT\_ID found in UNIT\_CONCEPT\_ID and in the ‘Unit’ domain within the Standardized Vocabularies. The original code is retained here for reference purposes. | varchar(50) | No | No | No |  |  |
| **unit\_source\_concept\_id** | “This is the concept representing the UNIT\_SOURCE\_VALUE and may not necessarily be standard. This field is discouraged from use in analysis because it is not required to contain Standard Concepts that are used across the OHDSI community, and should only be used when Standard Concepts do not adequately represent the source detail for the Measurement necessary for a given analytic use case. Consider using UNIT\_CONCEPT\_ID instead to enable standardized analytics that can be consistent across the network.” | If the UNIT\_SOURCE\_VALUE is coded in the source data using an OMOP supported vocabulary put the concept id representing the source value here. | integer | No | No | Yes | CONCEPT |  |
| **value\_source\_value** | This field houses the verbatim result value of the Measurement from the source data . | If both a continuous and categorical result are given in the source data such that both VALUE\_AS\_NUMBER and VALUE\_AS\_CONCEPT\_ID are both included, store the verbatim value that was mapped to VALUE\_AS\_CONCEPT\_ID here. | varchar(50) | No | No | No |  |  |
| **measurement\_event\_id** | If the Measurement record is related to another record in the database, this field is the primary key of the linked record. | Put the primary key of the linked record, if applicable, here. | integer | No | No | No |  |  |
| **meas\_event\_field\_concept\_id** | If the Measurement record is related to another record in the database, this field is the CONCEPT\_ID that identifies which table the primary key of the linked record came from. | Put the CONCEPT\_ID that identifies which table and field the MEASUREMENT\_EVENT\_ID came from. | integer | No | No | Yes | CONCEPT |  |

### observation

**Table Description**

The OBSERVATION table captures clinical facts about a Person obtained in the context of examination, questioning or a procedure. Any data that cannot be represented by any other domains, such as social and lifestyle facts, medical history, family history, etc. are recorded here.

**User Guide**

Observations differ from Measurements in that they do not require a standardized test or some other activity to generate clinical fact. Typical observations are medical history, family history, the stated need for certain treatment, social circumstances, lifestyle choices, healthcare utilization patterns, etc. If the generation clinical facts requires a standardized testing such as lab testing or imaging and leads to a standardized result, the data item is recorded in the MEASUREMENT table. If the clinical fact observed determines a sign, symptom, diagnosis of a disease or other medical condition, it is recorded in the CONDITION\_OCCURRENCE table. Valid Observation Concepts are not enforced to be from any domain but they must not belong to the Condition, Procedure, Drug, Device, Specimen, or Measurement domains and they must be Standard Concepts.  
  
The observation table usually records the date or datetime of when the observation was obtained, not the date of the observation starting. For example, if the patient reports that they had a heart attack when they were 50, the observation date or datetime is the date of the report, the heart attack observation can have a value\_as\_concept which captures how long ago the observation applied to the patient.

**ETL Conventions**

Records whose Source Values map to any domain besides Condition, Procedure, Drug, Specimen, Measurement or Device should be stored in the Observation table. Observations can be stored as attribute value pairs, with the attribute as the Observation Concept and the value representing the clinical fact. This fact can be a Concept (stored in VALUE\_AS\_CONCEPT), a numerical value (VALUE\_AS\_NUMBER), a verbatim string (VALUE\_AS\_STRING), or a datetime (VALUE\_AS\_DATETIME). Even though Observations do not have an explicit result, the clinical fact can be stated separately from the type of Observation in the VALUE\_AS\_\* fields. It is recommended for Observations that are suggestive statements of positive assertion should have a value of ‘Yes’ (concept\_id=4188539), recorded, even though the null value is the equivalent.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **observation\_id** | The unique key given to an Observation record for a Person. Refer to the ETL for how duplicate Observations during the same Visit were handled. | Each instance of an observation present in the source data should be assigned this unique key. | integer | Yes | Yes | No |  |  |
| **person\_id** | The PERSON\_ID of the Person for whom the Observation is recorded. This may be a system generated code. |  | integer | Yes | No | Yes | PERSON |  |
| **observation\_concept\_id** | The OBSERVATION\_CONCEPT\_ID field is recommended for primary use in analyses, and must be used for network studies. | The CONCEPT\_ID that the OBSERVATION\_SOURCE\_CONCEPT\_ID maps to. There is no specified domain that the Concepts in this table must adhere to. The only rule is that records with Concepts in the Condition, Procedure, Drug, Measurement, or Device domains MUST go to the corresponding table. | integer | Yes | No | Yes | CONCEPT |  |
| **observation\_date** | The date of when the Observation was obtained. Depending on what the Observation represents this could be the date of a lab test, the date of a survey, or the date a patient’s family history was taken. | For some observations the ETL may need to make a choice as to which date to choose. | date | Yes | No | No |  |  |
| **observation\_datetime** |  | If no time is given set to midnight (00:00:00). | datetime | No | No | No |  |  |
| **observation\_type\_concept\_id** | This field can be used to determine the provenance of the Observation record, as in whether the measurement was from an EHR system, insurance claim, registry, or other sources. | Choose the OBSERVATION\_TYPE\_CONCEPT\_ID that best represents the provenance of the record, for example whether it came from an EHR record or billing claim. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |
| **value\_as\_number** | This is the numerical value of the Result of the Observation, if applicable and available. It is not expected that all Observations will have numeric results, rather, this field is here to house values should they exist. |  | float | No | No | No |  |  |
| **value\_as\_string** | This is the categorical value of the Result of the Observation, if applicable and available. |  | varchar(60) | No | No | No |  |  |
| **value\_as\_concept\_id** | It is possible that some records destined for the Observation table have two clinical ideas represented in one source code. This is common with ICD10 codes that describe a family history of some Condition, for example. In OMOP the Vocabulary breaks these two clinical ideas into two codes; one becomes the OBSERVATION\_CONCEPT\_ID and the other becomes the VALUE\_AS\_CONCEPT\_ID. It is important when using the Observation table to keep this possibility in mind and to examine the VALUE\_AS\_CONCEPT\_ID field for relevant information. | Note that the value of VALUE\_AS\_CONCEPT\_ID may be provided through mapping from a source Concept which contains the content of the Observation. In those situations, the CONCEPT\_RELATIONSHIP table in addition to the ‘Maps to’ record contains a second record with the relationship\_id set to ‘Maps to value’. For example, ICD10 [Z82.4](https://athena.ohdsi.org/search-terms/terms/45581076) ‘Family history of ischaemic heart disease and other diseases of the circulatory system’ has a ‘Maps to’ relationship to [4167217](https://athena.ohdsi.org/search-terms/terms/4167217) ‘Family history of clinical finding’ as well as a ‘Maps to value’ record to [134057](https://athena.ohdsi.org/search-terms/terms/134057) ‘Disorder of cardiovascular system’. If there’s no categorial result in a source\_data, set value\_as\_concept\_id to NULL, if there is a categorial result in a source\_data but without mapping, set value\_as\_concept\_id to 0. | Integer | No | No | Yes | CONCEPT |  |
| **qualifier\_concept\_id** | This field contains all attributes specifying the clinical fact further, such as as degrees, severities, drug-drug interaction alerts etc. | Use your best judgement as to what Concepts to use here and if they are necessary to accurately represent the clinical record. There is no restriction on the domain of these Concepts, they just need to be Standard. | integer | No | No | Yes | CONCEPT |  |
| **unit\_concept\_id** | There is currently no recommended unit for individual observation concepts. UNIT\_SOURCE\_VALUES should be mapped to a Standard Concept in the Unit domain that best represents the unit as given in the source data. | There is no standardization requirement for units associated with OBSERVATION\_CONCEPT\_IDs, however, it is the responsibility of the ETL to choose the most plausible unit. If the source unit is NULL (applicable to cases when there’s no numerical value or when it doesn’t require a unit), keep unit\_concept\_id NULL as well. If there’s no mapping of a source unit, populate unit\_concept\_id with 0. | integer | No | No | Yes | CONCEPT | Unit |
| **provider\_id** | The provider associated with the observation record, e.g. the provider who ordered the test or the provider who recorded the result. | The ETL may need to make a choice as to which PROVIDER\_ID to put here. Based on what is available this may or may not be different than the provider associated with the overall VISIT\_OCCURRENCE record. For example the admitting vs attending physician on an EHR record. | integer | No | No | Yes | PROVIDER |  |
| **visit\_occurrence\_id** | The visit during which the Observation occurred. | Depending on the structure of the source data, this may have to be determined based on dates. If an OBSERVATION\_DATE occurs within the start and end date of a Visit it is a valid ETL choice to choose the VISIT\_OCCURRENCE\_ID from the visit that subsumes it, even if not explicitly stated in the data. While not required, an attempt should be made to locate the VISIT\_OCCURRENCE\_ID of the observation record. If an observation is related to a visit explicitly in the source data, it is possible that the result date of the Observation falls outside of the bounds of the Visit dates. | integer | No | No | Yes | VISIT\_OCCURRENCE |  |
| **visit\_detail\_id** | The VISIT\_DETAIL record during which the Observation occurred. For example, if the Person was in the ICU at the time the VISIT\_OCCURRENCE record would reflect the overall hospital stay and the VISIT\_DETAIL record would reflect the ICU stay during the hospital visit. | Same rules apply as for the VISIT\_OCCURRENCE\_ID. | integer | No | No | Yes | VISIT\_DETAIL |  |
| **observation\_source\_value** | This field houses the verbatim value from the source data representing the Observation that occurred. For example, this could be an ICD10 or Read code. | This code is mapped to a Standard Concept in the Standardized Vocabularies and the original code is stored here for reference. | varchar(50) | No | No | No |  |  |
| **observation\_source\_concept\_id** | This is the concept representing the OBSERVATION\_SOURCE\_VALUE and may not necessarily be standard. This field is discouraged from use in analysis because it is not required to contain Standard Concepts that are used across the OHDSI community, and should only be used when Standard Concepts do not adequately represent the source detail for the Observation necessary for a given analytic use case. Consider using OBSERVATION\_CONCEPT\_ID instead to enable standardized analytics that can be consistent across the network. | If the OBSERVATION\_SOURCE\_VALUE is coded in the source data using an OMOP supported vocabulary put the concept id representing the source value here. | integer | No | No | Yes | CONCEPT |  |
| **unit\_source\_value** | This field houses the verbatim value from the source data representing the unit of the Observation that occurred. | This code is mapped to a Standard Condition Concept in the Standardized Vocabularies and the original code is stored here for reference. | varchar(50) | No | No | No |  |  |
| **qualifier\_source\_value** | This field houses the verbatim value from the source data representing the qualifier of the Observation that occurred. | This code is mapped to a Standard Condition Concept in the Standardized Vocabularies and the original code is stored here for reference. | varchar(50) | No | No | No |  |  |
| **value\_source\_value** | This field houses the verbatim result value of the Observation from the source data. Do not get confused with the Observation\_source\_value which captures source value of the observation mapped to observation\_concept\_id. This field is the observation result value from the source. | If the observation\_source\_value was a question, for example, or an observation that requires a result then this field is the answer/ result from the source data. Store the verbatim value that represents the result of the observation\_source\_value. | varchar(50) | No | No | No |  |  |
| **observation\_event\_id** | If the Observation record is related to another record in the database, this field is the primary key of the linked record. | Put the primary key of the linked record, if applicable, here. See the [ETL Conventions for the OBSERVATION](https://ohdsi.github.io/CommonDataModel/cdm60.html#observation) table for more details. | integer | No | No | No |  |  |
| **obs\_event\_field\_concept\_id** | If the Observation record is related to another record in the database, this field is the CONCEPT\_ID that identifies which table the primary key of the linked record came from. | Put the CONCEPT\_ID that identifies which table and field the OBSERVATION\_EVENT\_ID came from. | integer | No | No | Yes | CONCEPT |  |

### death

**Table Description**

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 system contains evidence about the Death, such as: Condition in an administrative claim, status of enrollment into a health plan, or explicit record in EHR data.

**User Guide**

NA

**ETL Conventions**

For specific conventions on how to populate this table, please refer to the [THEMIS repository](https://ohdsi.github.io/Themis/death.html).

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **person\_id** |  |  | integer | Yes | No | Yes | PERSON |  |
| **death\_date** | The date the person was deceased. | If the precise date include 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. For additional conventions related to this field, please refer to the [THEMIS repository](https://ohdsi.github.io/Themis/tag_death_date.html). | date | Yes | No | No |  |  |
| **death\_datetime** |  | If you have date and time of death, populate death\_datetime, otherwise leave NULL | datetime | No | No | No |  |  |
| **death\_type\_concept\_id** | This is the provenance of the death record, i.e., where it came from. It is possible that an administrative claims database would source death information from a government file so do not assume the Death Type is the same as the Visit Type, etc. | Use the type concept that reflects the source of the death record. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | No | No | Yes | CONCEPT | Type Concept |
| **cause\_concept\_id** | This is the Standard Concept representing the Person’s cause of death, if available. | There is no specified domain for this concept, just choose the Standard Concept Id that best represents the person’s cause of death. | integer | No | No | Yes | CONCEPT |  |
| **cause\_source\_value** |  | If available, put the source code representing the cause of death here. | varchar(50) | No | No | No |  |  |
| **cause\_source\_concept\_id** |  | If the cause of death was coded using a Vocabulary present in the OMOP Vocabularies (not necessarily a standard concept) put the CONCEPT\_ID representing the cause of death here. | integer | No | No | Yes | CONCEPT |  |

### note

**Table Description**

The NOTE table captures unstructured information that was recorded by a provider about a patient in free text (in ASCII, or preferably in UTF8 format) notes on a given date. The type of note\_text is CLOB or varchar(MAX) depending on RDBMS.

**User Guide**

NA

**ETL Conventions**

HL7/LOINC CDO is a standard for consistent naming of documents to support a range of use cases: retrieval, organization, display, and exchange. It guides the creation of LOINC codes for clinical notes. CDO annotates each document with 5 dimensions:

* **Kind of Document**: Characterizes the general structure of the document at a macro level (e.g. Anesthesia Consent)
* **Type of Service**: Characterizes the kind of service or activity (e.g. evaluations, consultations, and summaries). The notion of time sequence, e.g., at the beginning (admission) at the end (discharge) is subsumed in this axis. Example: Discharge Teaching.
* **Setting**: Setting is an extension of CMS’s definitions (e.g. Inpatient, Outpatient)
* **Subject Matter Domain (SMD)**: Characterizes the subject matter domain of a note (e.g. Anesthesiology)
* **Role**: Characterizes the training or professional level of the author of the document, but does not break down to specialty or subspecialty (e.g. Physician) Each combination of these 5 dimensions rolls up to a unique LOINC code.

According to CDO requirements, only 2 of the 5 dimensions are required to properly annotate a document; Kind of Document and any one of the other 4 dimensions. However, not all the permutations of the CDO dimensions will necessarily yield an existing LOINC code. Each of these dimensions are contained in the OMOP Vocabulary under the domain of ‘Meas Value’ with each dimension represented as a Concept Class.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **note\_id** | A unique identifier for each note. |  | integer | Yes | Yes | No |  |  |
| **person\_id** |  |  | integer | Yes | No | Yes | PERSON |  |
| **note\_date** | The date the note was recorded. |  | date | Yes | No | No |  |  |
| **note\_datetime** |  | If time is not given set the time to midnight. | datetime | No | No | No |  |  |
| **note\_type\_concept\_id** | The provenance of the note. Most likely this will be EHR. | Put the source system of the note, as in EHR record. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?standardConcept=Standard&domain=Type+Concept&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |
| **note\_class\_concept\_id** | A Standard Concept Id representing the HL7 LOINC Document Type Vocabulary classification of the note. | Map the note classification to a Standard Concept. For more information see the ETL Conventions in the description of the NOTE table. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?standardConcept=Standard&conceptClass=Doc+Kind&conceptClass=Doc+Role&conceptClass=Doc+Setting&conceptClass=Doc+Subject+Matter&conceptClass=Doc+Type+of+Service&domain=Meas+Value&page=1&pageSize=15&query=). This Concept can alternatively be represented by concepts with the relationship ‘Kind of (LOINC)’ to [706391](https://athena.ohdsi.org/search-terms/terms/706391) (Note). | integer | Yes | No | Yes | CONCEPT |  |
| **note\_title** | The title of the note. |  | varchar(250) | No | No | No |  |  |
| **note\_text** | The content of the note. |  | varchar(MAX) | Yes | No | No |  |  |
| **encoding\_concept\_id** | This is the Concept representing the character encoding type. | Put the Concept Id that represents the encoding character type here. Currently the only option is UTF-8 ([32678](https://athena.ohdsi.org/search-terms/terms/32678)). It the note is encoded in any other type, like ASCII then put 0. | integer | Yes | No | Yes | CONCEPT |  |
| **language\_concept\_id** | The language of the note. | Use Concepts that are descendants of the concept [4182347](https://athena.ohdsi.org/search-terms/terms/4182347) (World Languages). | integer | Yes | No | Yes | CONCEPT |  |
| **provider\_id** | The Provider who wrote the note. | The ETL may need to make a determination on which provider to put here. | integer | No | No | Yes | PROVIDER |  |
| **visit\_occurrence\_id** | The Visit during which the note was written. |  | integer | No | No | Yes | VISIT\_OCCURRENCE |  |
| **visit\_detail\_id** | The Visit Detail during which the note was written. |  | integer | No | No | Yes | VISIT\_DETAIL |  |
| **note\_source\_value** |  | The source value mapped to the NOTE\_CLASS\_CONCEPT\_ID. | varchar(50) | No | No | No |  |  |
| **note\_event\_id** | If the Note record is related to another record in the database, this field is the primary key of the linked record. | Put the primary key of the linked record, if applicable, here. | integer | No | No | No |  |  |
| **note\_event\_field\_concept\_id** | If the Note record is related to another record in the database, this field is the CONCEPT\_ID that identifies which table the primary key of the linked record came from. | Put the CONCEPT\_ID that identifies which table and field the NOTE\_EVENT\_ID came from. | integer | No | No | Yes | CONCEPT |  |

### note\_nlp

**Table Description**

The NOTE\_NLP table encodes all output of NLP on clinical notes. Each row represents a single extracted term from a note.

**User Guide**

NA

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **note\_nlp\_id** | A unique identifier for the NLP record. |  | integer | Yes | Yes | No |  |  |
| **note\_id** | This is the NOTE\_ID for the NOTE record the NLP record is associated to. |  | integer | Yes | No | No |  |  |
| **section\_concept\_id** |  | The SECTION\_CONCEPT\_ID should be used to represent the note section contained in the NOTE\_NLP record. These concepts can be found as parts of document panels and are based on the type of note written, i.e. a discharge summary. These panels can be found as concepts with the relationship ‘Subsumes’ to CONCEPT\_ID [45875957](https://athena.ohdsi.org/search-terms/terms/45875957). | integer | No | No | Yes | CONCEPT |  |
| **snippet** | A small window of text surrounding the term |  | varchar(250) | No | No | No |  |  |
| **“offset”** | Character offset of the extracted term in the input note |  | varchar(50) | No | No | No |  |  |
| **lexical\_variant** | Raw text extracted from the NLP tool. |  | varchar(250) | Yes | No | No |  |  |
| **note\_nlp\_concept\_id** |  |  | integer | No | No | Yes | CONCEPT |  |
| **note\_nlp\_source\_concept\_id** |  |  | integer | No | No | Yes | CONCEPT |  |
| **nlp\_system** |  | Name and version of the NLP system that extracted the term. Useful for data provenance. | varchar(250) | No | No | No |  |  |
| **nlp\_date** | The date of the note processing. |  | date | Yes | No | No |  |  |
| **nlp\_datetime** | The date and time of the note processing. |  | datetime | No | No | No |  |  |
| **term\_exists** |  | Term\_exists is defined as a flag that indicates if the patient actually has or had the condition. Any of the following modifiers would make Term\_exists false: Negation = true Subject = [anything other than the patient] Conditional = true/li> Rule\_out = true Uncertain = very low certainty or any lower certainties A complete lack of modifiers would make Term\_exists true. | varchar(1) | No | No | No |  |  |
| **term\_temporal** |  | Term\_temporal is to indicate if a condition is present or just in the past. The following would be past:  - History = true - Concept\_date = anything before the time of the report | varchar(50) | No | No | No |  |  |
| **term\_modifiers** |  | For the modifiers that are there, they would have to have these values:  - Negation = false - Subject = patient - Conditional = false - Rule\_out = false - Uncertain = true or high or moderate or even low (could argue about low). Term\_modifiers will concatenate all modifiers for different types of entities (conditions, drugs, labs etc) into one string. Lab values will be saved as one of the modifiers. | varchar(2000) | No | No | No |  |  |

### specimen

**Table Description**

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

**User Guide**

NA

**ETL 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.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **specimen\_id** | Unique identifier for each specimen. |  | integer | Yes | Yes | No |  |  |
| **person\_id** | The person from whom the specimen is collected. |  | integer | Yes | No | Yes | PERSON |  |
| **specimen\_concept\_id** |  | The standard CONCEPT\_ID that the SPECIMEN\_SOURCE\_VALUE maps to in the specimen domain. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Specimen&standardConcept=Standard&page=1&pageSize=15&query=) | integer | Yes | No | Yes | CONCEPT |  |
| **specimen\_type\_concept\_id** |  | Put the source of the specimen record, as in an EHR system. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?standardConcept=Standard&domain=Type+Concept&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |
| **specimen\_date** | The date the specimen was collected. |  | date | Yes | No | No |  |  |
| **specimen\_datetime** |  |  | datetime | No | No | No |  |  |
| **quantity** | The amount of specimen collected from the person. |  | float | No | No | No |  |  |
| **unit\_concept\_id** | The unit for the quantity of the specimen. | Map the UNIT\_SOURCE\_VALUE to a Standard Concept in the Unit domain. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Unit&standardConcept=Standard&page=1&pageSize=15&query=). If the source unit is NULL (applicable to cases when there’s no numerical value or when it doesn’t require a unit), keep unit\_concept\_id NULL as well. If there’s no mapping of a source unit, populate unit\_concept\_id with 0. | integer | No | No | Yes | CONCEPT |  |
| **anatomic\_site\_concept\_id** | This is the site on the body where the specimen is from. | Map the ANATOMIC\_SITE\_SOURCE\_VALUE to a Standard Concept in the Spec Anatomic Site domain. This should be coded at the lowest level of granularity [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?standardConcept=Standard&domain=Spec+Anatomic+Site&conceptClass=Body+Structure&page=4&pageSize=15&query=) | integer | No | No | Yes | CONCEPT |  |
| **disease\_status\_concept\_id** |  |  | integer | No | No | Yes | CONCEPT |  |
| **specimen\_source\_id** | This is the identifier for the specimen from the source system. |  | varchar(50) | No | No | No |  |  |
| **specimen\_source\_value** |  |  | varchar(50) | No | No | No |  |  |
| **unit\_source\_value** |  | This unit for the quantity of the specimen, as represented in the source. | varchar(50) | No | No | No |  |  |
| **anatomic\_site\_source\_value** |  | This is the site on the body where the specimen was taken from, as represented in the source. | varchar(50) | No | No | No |  |  |
| **disease\_status\_source\_value** |  |  | varchar(50) | No | No | No |  |  |

### fact\_relationship

**Table Description**

The FACT\_RELATIONSHIP table contains records about the relationships between facts stored as records in any table of the CDM. Relationships can be defined between facts from the same domain, or different domains. Examples of Fact Relationships include: [Person relationships](https://athena.ohdsi.org/search-terms/terms?domain=Relationship&standardConcept=Standard&page=2&pageSize=15&query=) (parent-child), care site relationships (hierarchical organizational structure of facilities within a health system), indication relationship (between drug exposures and associated conditions), usage relationships (of devices during the course of an associated procedure), or facts derived from one another (measurements derived from an associated specimen).

**User Guide**

NA

**ETL Conventions**

All relationships are directional, and each relationship is represented twice symmetrically within the FACT\_RELATIONSHIP table. For example, two persons if person\_id = 1 is the mother of person\_id = 2 two records are in the FACT\_RELATIONSHIP table (all strings in fact concept\_id records in the Concept table: - Person, 1, Person, 2, parent of - Person, 2, Person, 1, child of

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **domain\_concept\_id\_1** |  |  | integer | Yes | No | Yes | CONCEPT |  |
| **fact\_id\_1** |  |  | integer | Yes | No | No |  |  |
| **domain\_concept\_id\_2** |  |  | integer | Yes | No | Yes | CONCEPT |  |
| **fact\_id\_2** |  |  | integer | Yes | No | No |  |  |
| **relationship\_concept\_id** |  |  | integer | Yes | No | Yes | CONCEPT |  |

### location

**Table Description**

The LOCATION table represents a generic way to capture physical location or address information of Persons and Care Sites.

**User Guide**

The current iteration of the LOCATION table is US centric. Until a major release to correct this, certain fields can be used to represent different international values.  
  
- STATE can also be used for province or district  
- ZIP is also the postal code or postcode  
- COUNTY can also be used to represent region

**ETL Conventions**

Each address or Location is unique and is present only once in the table. Locations do not contain names, such as the name of a hospital. In order to construct a full address that can be used in the postal service, the address information from the Location needs to be combined with information from the Care Site.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **location\_id** | The unique key given to a unique Location. | Each instance of a Location in the source data should be assigned this unique key. | integer | Yes | Yes | No |  |  |
| **address\_1** | This is the first line of the address. |  | varchar(50) | No | No | No |  |  |
| **address\_2** | This is the second line of the address |  | varchar(50) | No | No | No |  |  |
| **city** |  |  | varchar(50) | No | No | No |  |  |
| **state** |  |  | varchar(2) | No | No | No |  |  |
| **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, the full 5-digit Zip or the 9-digit (ZIP + 4) codes. Unless for specific reasons analytical methods should expect and utilize only the first 3 digits. For international addresses, different rules apply. | varchar(9) | No | No | No |  |  |
| **county** |  |  | varchar(20) | No | No | No |  |  |
| **location\_source\_value** |  | Put the verbatim value for the location here, as it shows up in the source. | varchar(50) | No | No | No |  |  |
| **country\_concept\_id** | The Concept Id representing the country. Values should conform to the [Geography](https://athena.ohdsi.org/search-terms/terms?domain=Geography&standardConcept=Standard&page=1&pageSize=15&query=&boosts) domain. |  | integer | No | No | Yes | CONCEPT |  |
| **country\_source\_value** | The name of the country. |  | varchar(80) | No | No | No |  |  |
| **latitude** |  | Must be between -90 and 90. | float | No | No | No |  |  |
| **longitude** |  | Must be between -180 and 180. | float | No | No | No |  |  |

### care\_site

**Table Description**

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

**User Guide**

NA

**ETL Conventions**

Care site is a unique combination of location\_id and nature of the site - the latter could be the place of service, name, or another characteristic in your source data. Care site does not take into account the provider (human) information such a specialty. Many source data do not make a distinction between individual and institutional providers. The CARE\_SITE table contains the institutional providers. If the source, instead of uniquely identifying individual Care Sites, only provides limited information such as Place of Service, generic or “pooled” Care Site records are listed in the CARE\_SITE table. There can be hierarchical and business relationships between Care Sites. For example, wards can belong to clinics or departments, which can in turn belong to hospitals, which in turn can belong to hospital systems, which in turn can belong to HMOs.The relationships between Care Sites are defined in the FACT\_RELATIONSHIP table.  
  
For additional detailed conventions on how to populate this table, please refer to [THEMIS repository](https://ohdsi.github.io/Themis/care_site.html).

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **care\_site\_id** |  | Assign an ID to each combination of a location and nature of the site - the latter could be the Place of Service, name or another characteristic in your source data. | integer | Yes | Yes | No |  |  |
| **care\_site\_name** | The name of the care\_site as it appears in the source data |  | varchar(255) | No | No | No |  |  |
| **place\_of\_service\_concept\_id** | This is a high-level way of characterizing a Care Site. Typically, however, Care Sites can provide care in multiple settings (inpatient, outpatient, etc.) and this granularity should be reflected in the visit. | Choose the concept in the visit domain that best represents the setting in which healthcare is provided in the Care Site. If most visits in a Care Site are Inpatient, then the place\_of\_service\_concept\_id should represent Inpatient. If information is present about a unique Care Site (e.g. Pharmacy) then a Care Site record should be created. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Visit&standardConcept=Standard&page=2&pageSize=15&query=). For information about how to populate this field please see the [THEMIS Conventions](https://ohdsi.github.io/Themis/tag_place_of_service.html). | integer | No | No | Yes | CONCEPT |  |
| **location\_id** | The location\_id from the LOCATION table representing the physical location of the care\_site. |  | integer | No | No | Yes | LOCATION |  |
| **care\_site\_source\_value** | The identifier of the care\_site as it appears in the source data. This could be an identifier separate from the name of the care\_site. |  | varchar(50) | No | No | No |  |  |
| **place\_of\_service\_source\_value** |  | Put the place of service of the care\_site as it appears in the source data. | varchar(50) | No | No | No |  |  |

### provider

**Table Description**

The PROVIDER table contains a list of uniquely identified healthcare providers; duplication is not allowed. These are individuals providing hands-on healthcare to patients, such as physicians, nurses, midwives, physical therapists etc.

**User Guide**

Many sources do not make a distinction between individual and institutional providers. The PROVIDER table contains the individual providers. If the source only provides limited information such as specialty instead of uniquely identifying individual providers, generic or ‘pooled’ Provider records are listed in the PROVIDER table.

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **provider\_id** | It is assumed that every provider with a different unique identifier is in fact a different person and should be treated independently. | This identifier can be the original id from the source data provided it is an integer, otherwise it can be an autogenerated number. | integer | Yes | Yes | No |  |  |
| **provider\_name** | This field contains information that describes a healthcare provider. | This field is not required for identifying the Provider’s actual identity. Instead, its purpose is to uniquely and/or anonymously identify providers of care across the database. | varchar(255) | No | No | No |  |  |
| **npi** | This is the National Provider Number issued to health care providers in the US by the Centers for Medicare and Medicaid Services (CMS). |  | varchar(20) | No | No | No |  |  |
| **dea** | This is the identifier issued by the DEA, a US federal agency, that allows a provider to write prescriptions for controlled substances. |  | varchar(20) | No | No | No |  |  |
| **specialty\_concept\_id** | This field either represents the most common specialty that occurs in the data or the most specific concept that represents all specialties listed, should the provider have more than one. This includes physician specialties such as internal medicine, emergency medicine, etc. and allied health professionals such as nurses, midwives, and pharmacists. | If a Provider has more than one Specialty, there are two options: 1. Choose a concept\_id which is a common ancestor to the multiple specialties, or, 2. Choose the specialty that occurs most often for the provider. Concepts in this field should be Standard with a domain of Provider. [Accepted Concepts](http://athena.ohdsi.org/search-terms/terms?domain=Provider&standardConcept=Standard&page=1&pageSize=15&query=). | integer | No | No | Yes | CONCEPT |  |
| **care\_site\_id** | This is the CARE\_SITE\_ID for the location that the provider primarily practices in. | If a Provider has more than one Care Site, the main or most often exerted CARE\_SITE\_ID should be recorded. | integer | No | No | Yes | CARE\_SITE |  |
| **year\_of\_birth** |  |  | integer | No | No | No |  |  |
| **gender\_concept\_id** | This field represents the recorded gender of the provider in the source data. | If given, put a concept from the gender domain representing the recorded gender of the provider. [Accepted Concepts](http://athena.ohdsi.org/search-terms/terms?domain=Gender&standardConcept=Standard&page=1&pageSize=15&query=). | integer | No | No | Yes | CONCEPT | Gender |
| **provider\_source\_value** | Use this field to link back to providers in the source data. This is typically used for error checking of ETL logic. | Some use cases require the ability to link back to providers in the source data. This field allows for the storing of the provider identifier as it appears in the source. | varchar(50) | No | No | No |  |  |
| **specialty\_source\_value** | This refers to the specific type of healthcare provider or field of expertise listed in the source data, encompassing physician specialties like internal medicine, emergency medicine, etc., as well as allied health professionals such as nurses, midwives, and pharmacists. It covers medical specialties like surgery, internal medicine, and radiology, while other services like prosthetics, acupuncture, and physical therapy fall under the domain of “Service.” | The type of provider and their specialty should be entered as they appear in the source data. The decision to use either the coded value or the text description is left to the discretion of the ETL-er. | varchar(50) | No | No | No |  |  |
| **specialty\_source\_concept\_id** | This is often zero as many sites use proprietary codes to store physician speciality. | If the source data codes provider specialty in an OMOP supported vocabulary store the concept\_id here. | integer | No | No | Yes | CONCEPT |  |
| **gender\_source\_value** | This is provider’s gender as it appears in the source data. | Put the provider’s gender as it appears in the source data. This field is up to the discretion of the ETL-er as to whether this should be the coded value from the source or the text description of the lookup value. | varchar(50) | No | No | No |  |  |
| **gender\_source\_concept\_id** | This is often zero as many sites use proprietary codes to store provider gender. | If the source data codes provider gender in an OMOP supported vocabulary store the concept\_id here. | integer | No | No | Yes | CONCEPT |  |

### payer\_plan\_period

**Table Description**

The PAYER\_PLAN\_PERIOD table captures details of the period of time that a Person is continuously enrolled under a specific health Plan benefit structure from a given Payer. Each Person receiving healthcare is typically covered by a health benefit plan, which pays for (fully or partially), or directly provides, the care. These benefit plans are provided by payers, such as health insurances or state or government agencies. In each plan the details of the health benefits are defined for the Person or her family, and the health benefit Plan might change over time typically with increasing utilization (reaching certain cost thresholds such as deductibles), plan availability and purchasing choices of the Person. The unique combinations of Payer organizations, health benefit Plans and time periods in which they are valid for a Person are recorded in this table.

**User Guide**

A Person can have multiple, overlapping, Payer\_Plan\_Periods in this table. For example, medical and drug coverage in the US can be represented by two Payer\_Plan\_Periods. The details of the benefit structure of the Plan is rarely known, the idea is just to identify that the Plans are different.

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **payer\_plan\_period\_id** | A unique identifier for each unique combination of a Person, Payer, Plan, and Period of time. |  | integer | Yes | Yes | No |  |  |
| **person\_id** | The Person covered by the Plan. | A single Person can have multiple, overlapping, PAYER\_PLAN\_PERIOD records | integer | Yes | No | Yes | PERSON |  |
| **payer\_plan\_period\_start\_date** | Start date of Plan coverage. |  | date | Yes | No | No |  |  |
| **payer\_plan\_period\_end\_date** | End date of Plan coverage. |  | date | Yes | No | No |  |  |
| **payer\_concept\_id** | This field represents the organization who reimburses the provider which administers care to the Person. | Map the payer directly to a standard CONCEPT\_ID with the domain\_id of ‘Payer’ ([Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Payer&standardConcept=Standard&page=1&pageSize=15&query=)). This vocabulary is not exhaustive so if there is a value missing, please see the [custom concepts](https://ohdsi.github.io/CommonDataModel/customConcepts.html) page. | integer | No | No | Yes | CONCEPT |  |
| **payer\_source\_value** | This is the Payer as it appears in the source data. |  | varchar(50) | No | No | No |  |  |
| **payer\_source\_concept\_id** |  | If the source data codes the Payer in an OMOP supported vocabulary store the concept\_id here. | integer | No | No | Yes | CONCEPT |  |
| **plan\_concept\_id** | This field represents the specific health benefit Plan the Person is enrolled in. | Map the Plan directly to a standard CONCEPT\_ID in the ‘Plan’ vocabulary ([Accepted Concepts](http://athena.ohdsi.org/search-terms/terms?domain=Plan&standardConcept=Standard&page=1&pageSize=15&query=)). This vocabulary is not exhaustive so if there is a value missing, please see the [custom concepts](https://ohdsi.github.io/CommonDataModel/customConcepts.html) page. | integer | No | No | Yes | CONCEPT |  |
| **plan\_source\_value** | This is the health benefit Plan of the Person as it appears in the source data. |  | varchar(50) | No | No | No |  |  |
| **plan\_source\_concept\_id** |  | If the source data codes the Plan in an OMOP supported vocabulary store the concept\_id here. | integer | No | No | Yes | CONCEPT |  |
| **sponsor\_concept\_id** | This field represents the sponsor of the Plan who finances the Plan. This includes self-insured, small group health plan and large group health plan. | Map the sponsor directly to a standard CONCEPT\_ID with the domain\_id of ‘Sponsor’ ([Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Sponsor&standardConcept=Standard&page=1&pageSize=15&query=)). This vocabulary is not exhaustive so if there is a value missing, please see the [custom concepts](https://ohdsi.github.io/CommonDataModel/customConcepts.html) page. | integer | No | No | Yes | CONCEPT |  |
| **sponsor\_source\_value** | The Plan sponsor as it appears in the source data. |  | varchar(50) | No | No | No |  |  |
| **sponsor\_source\_concept\_id** |  | If the source data codes the sponsor in an OMOP supported vocabulary store the concept\_id here. | integer | No | No | Yes | CONCEPT |  |
| **family\_source\_value** | The common identifier for all people (often a family) that covered by the same policy. | Often these are the common digits of the enrollment id of the policy members. | varchar(50) | No | No | No |  |  |
| **stop\_reason\_concept\_id** | This field represents the reason the Person left the Plan, if known. | Map the stop reason directly to a standard CONCEPT\_ID with a domain of ‘Plan Stop Reason’ ([Accepted Concepts](http://athena.ohdsi.org/search-terms/terms?domain=Plan+Stop+Reason&standardConcept=Standard&page=1&pageSize=15&query=)). If one does not exist visit the [Custom Concepts](https://ohdsi.github.io/CommonDataModel/customConcepts.html) pate for more information. | integer | No | No | Yes | CONCEPT |  |
| **stop\_reason\_source\_value** | The Plan stop reason as it appears in the source data. |  | varchar(50) | No | No | No |  |  |
| **stop\_reason\_source\_concept\_id** |  | If the source data codes the stop reason in an OMOP supported vocabulary store the concept\_id here. | integer | No | No | Yes | CONCEPT |  |

### cost

**Table Description**

The COST table captures records containing the cost of any medical event recorded in one of the OMOP clinical event tables such as DRUG\_EXPOSURE, PROCEDURE\_OCCURRENCE, VISIT\_OCCURRENCE, VISIT\_DETAIL, DEVICE\_OCCURRENCE, OBSERVATION or MEASUREMENT.

Each record in the cost table account for the amount of money transacted for the clinical event. So, the COST table may be used to represent both receivables (charges) and payments (paid), each transaction type represented by its COST\_CONCEPT\_ID. The COST\_TYPE\_CONCEPT\_ID field will use concepts in the Standardized Vocabularies to designate the source (provenance) of the cost data. A reference to the health plan information in the PAYER\_PLAN\_PERIOD table is stored in the record for information used for the adjudication system to determine the persons benefit for the clinical event.

**User Guide**

When dealing with summary costs, the cost of the goods or services the provider provides is often not known directly, but derived from the hospital charges multiplied by an average cost-to-charge ratio.

**ETL Conventions**

One cost record is generated for each response by a payer. In a claims databases, the payment and payment terms reported by the payer for the goods or services billed will generate one cost record. If the source data has payment information for more than one payer (i.e. primary insurance and secondary insurance payment for one entity), then a cost record is created for each reporting payer. Therefore, it is possible for one procedure to have multiple cost records for each payer, but typically it contains one or no record per entity. Payer reimbursement cost records will be identified by using the PAYER\_PLAN\_ID field. Drug costs are composed of ingredient cost (the amount charged by the wholesale distributor or manufacturer), the dispensing fee (the amount charged by the pharmacy and the sales tax).

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **cost\_id** |  |  | integer | Yes | Yes | No |  |  |
| **cost\_event\_id** |  |  | integer | Yes | No | No |  |  |
| **cost\_domain\_id** |  |  | varchar(20) | Yes | No | Yes | DOMAIN |  |
| **cost\_type\_concept\_id** |  |  | integer | Yes | No | Yes | CONCEPT |  |
| **currency\_concept\_id** |  |  | integer | No | No | Yes | CONCEPT |  |
| **total\_charge** |  |  | float | No | No | No |  |  |
| **total\_cost** |  |  | float | No | No | No |  |  |
| **total\_paid** |  |  | float | No | No | No |  |  |
| **paid\_by\_payer** |  |  | float | No | No | No |  |  |
| **paid\_by\_patient** |  |  | float | No | No | No |  |  |
| **paid\_patient\_copay** |  |  | float | No | No | No |  |  |
| **paid\_patient\_coinsurance** |  |  | float | No | No | No |  |  |
| **paid\_patient\_deductible** |  |  | float | No | No | No |  |  |
| **paid\_by\_primary** |  |  | float | No | No | No |  |  |
| **paid\_ingredient\_cost** |  |  | float | No | No | No |  |  |
| **paid\_dispensing\_fee** |  |  | float | No | No | No |  |  |
| **payer\_plan\_period\_id** |  |  | integer | No | No | No |  |  |
| **amount\_allowed** |  |  | float | No | No | No |  |  |
| **revenue\_code\_concept\_id** |  |  | integer | No | No | Yes | CONCEPT |  |
| **revenue\_code\_source\_value** | Revenue codes are a method to charge for a class of procedures and conditions in the U.S. hospital system. |  | varchar(50) | No | No | No |  |  |
| **drg\_concept\_id** |  |  | integer | No | No | Yes | CONCEPT |  |
| **drg\_source\_value** | Diagnosis Related Groups are US codes used to classify hospital cases into one of approximately 500 groups. |  | varchar(3) | No | No | No |  |  |

### drug\_era

**Table Description**

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. Every record in the DRUG\_EXPOSURE table should be part of a drug era based on the dates of exposure.

**User Guide**

NA

**ETL Conventions**

The SQL script for generating DRUG\_ERA records can be found [here](https://ohdsi.github.io/CommonDataModel/sqlScripts.html#drug_eras).

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **drug\_era\_id** |  |  | integer | Yes | Yes | No |  |  |
| **person\_id** |  |  | integer | Yes | No | Yes | PERSON |  |
| **drug\_concept\_id** | The drug\_concept\_id should conform to the concept class ‘ingredient’ as the drug\_era is an era of time where a person is exposed to a particular drug ingredient. |  | integer | Yes | No | Yes | CONCEPT | Drug |
| **drug\_era\_start\_date** |  | The Drug Era Start Date is the start date of the first Drug Exposure for a given ingredient, with at least 31 days since the previous exposure. | date | Yes | No | No |  |  |
| **drug\_era\_end\_date** |  | 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 prescription data, the date when the drug was dispensed plus the number of days of supply are used to extrapolate the End Date for the Drug Exposure. Depending on the country-specific healthcare system, this supply information is either explicitly provided in the day\_supply field or inferred from package size or similar information. For Procedure Drugs, usually the drug is administered on a single date (i.e., the administration date). A standard Persistence Window of 30 days (gap, slack) is permitted between two subsequent such extrapolated DRUG\_EXPOSURE records to be considered to be merged into a single Drug Era. | date | Yes | No | No |  |  |
| **drug\_exposure\_count** | The count of grouped DRUG\_EXPOSURE records that were included in the DRUG\_ERA row |  | integer | No | No | No |  |  |
| **gap\_days** |  | The Gap Days determine how many total drug-free days are observed between all Drug Exposure events that contribute to a DRUG\_ERA record. It is assumed that the drugs are “not stockpiled” by the patient, i.e. that if a new drug prescription or refill is observed (a new DRUG\_EXPOSURE record is written), the remaining supply from the previous events is abandoned. The difference between Persistence Window and Gap Days is that the former is the maximum drug-free time allowed between two subsequent DRUG\_EXPOSURE records, while the latter is the sum of actual drug-free days for the given Drug Era under the above assumption of non-stockpiling. | integer | No | No | No |  |  |

### dose\_era

**Table Description**

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.

**User Guide**

NA

**ETL Conventions**

Dose Eras will be derived from records in the DRUG\_EXPOSURE table and the Dose information from the DRUG\_STRENGTH table using a standardized algorithm. Dose Form information is not taken into account. So, if the patient changes between different formulations, or different manufacturers with the same formulation, the Dose Era is still spanning the entire time of exposure to the Ingredient.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **dose\_era\_id** |  |  | integer | Yes | Yes | No |  |  |
| **person\_id** |  |  | integer | Yes | No | Yes | PERSON |  |
| **drug\_concept\_id** | The Concept Id representing the specific drug ingredient. |  | integer | Yes | No | Yes | CONCEPT | Drug |
| **unit\_concept\_id** | The Concept Id representing the unit of the specific drug ingredient. |  | integer | Yes | No | Yes | CONCEPT | Unit |
| **dose\_value** | The numeric value of the dosage of the drug\_ingredient. |  | float | Yes | No | No |  |  |
| **dose\_era\_start\_date** | The date the Person started on the specific dosage, with at least 31 days since any prior exposure. |  | date | Yes | No | No |  |  |
| **dose\_era\_end\_date** |  | The date the Person was no longer exposed to the dosage of the specific drug ingredient. An era is ended if there are 31 days or more between dosage records. | date | Yes | No | No |  |  |

### condition\_era

**Table Description**

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 and every Condition Occurrence record should be part of a Condition Era. 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 referred to a specialist. At a later time, the Person visits the specialist, who confirms the PCP’s original diagnosis and provides the appropriate treatment to resolve the condition. These two independent doctor visits should be aggregated into one Condition Era.

**User Guide**

NA

**ETL Conventions**

Each Condition Era corresponds to one or many Condition Occurrence records that form a continuous interval. The condition\_concept\_id field contains Concepts that are identical to those of the CONDITION\_OCCURRENCE table records that make up the Condition Era. In contrast to Drug Eras, Condition Eras are not aggregated to contain Conditions of different hierarchical layers. The SQl Script for generating CONDITION\_ERA records can be found [here](https://ohdsi.github.io/CommonDataModel/sqlScripts.html#condition_eras) 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. Condition Eras are built with a Persistence Window of 30 days, meaning, if no occurrence of the same condition\_concept\_id happens within 30 days of any one occurrence, it will be considered the condition\_era\_end\_date.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **condition\_era\_id** |  |  | integer | Yes | Yes | No |  |  |
| **person\_id** |  |  | integer | Yes | No | Yes | PERSON |  |
| **condition\_concept\_id** | The Concept Id representing the Condition. |  | integer | Yes | No | Yes | CONCEPT | Condition |
| **condition\_era\_start\_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 with at least 31 days since any prior record of the same Condition. |  | date | Yes | No | No |  |  |
| **condition\_era\_end\_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. |  | date | Yes | No | No |  |  |
| **condition\_occurrence\_count** | The number of individual Condition Occurrences used to construct the condition era. |  | integer | No | No | No |  |  |

### episode

**Table Description**

The EPISODE table aggregates lower-level clinical events (VISIT\_OCCURRENCE, DRUG\_EXPOSURE, PROCEDURE\_OCCURRENCE, DEVICE\_EXPOSURE) into a higher-level abstraction representing clinically and analytically relevant disease phases,outcomes and treatments. The EPISODE\_EVENT table connects qualifying clinical events (VISIT\_OCCURRENCE, DRUG\_EXPOSURE, PROCEDURE\_OCCURRENCE, DEVICE\_EXPOSURE) to the appropriate EPISODE entry. For example cancers including their development over time, their treatment, and final resolution.

**User Guide**

Valid Episode Concepts belong to the ‘Episode’ domain. For cancer episodes please see [article], for non-cancer episodes please see [article]. If your source data does not have all episodes that are relevant to the therapeutic area, write only those you can easily derive from the data. It is understood that that table is not currently expected to be comprehensive.

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **episode\_id** | A unique identifier for each Episode. |  | integer | Yes | Yes | No |  |  |
| **person\_id** | The PERSON\_ID of the PERSON for whom the episode is recorded. |  | integer | Yes | No | Yes | PERSON |  |
| **episode\_concept\_id** | The EPISODE\_CONCEPT\_ID represents the kind abstraction related to the disease phase, outcome or treatment. | Choose a concept in the Episode domain that best represents the ongoing disease phase, outcome, or treatment. Please see [article] for cancers and [article] for non-cancers describing how these are defined. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Episode&page=1&pageSize=15&query=) | integer | Yes | No | Yes | CONCEPT | Episode |
| **episode\_start\_date** | The date when the Episode beings. | Please see [article] for how to define an Episode start date. | date | Yes | No | No |  |  |
| **episode\_start\_datetime** | The date and time when the Episode begins. |  | datetime | No | No | No |  |  |
| **episode\_end\_date** | The date when the instance of the Episode is considered to have ended. | Please see [article] for how to define an Episode end date. | date | No | No | No |  |  |
| **episode\_end\_datetime** | The date when the instance of the Episode is considered to have ended. |  | datetime | No | No | No |  |  |
| **episode\_parent\_id** | Use this field to find the Episode that subsumes the given Episode record. This is used in the case that an Episode are nested into each other. | If there are multiple nested levels to how Episodes are represented, the EPISODE\_PARENT\_ID can be used to record this relationship. | integer | No | No | No |  |  |
| **episode\_number** | For sequences of episodes, this is used to indicate the order the episodes occurred. For example, lines of treatment could be indicated here. | Please see [article] for the details of how to count episodes. | integer | No | No | No |  |  |
| **episode\_object\_concept\_id** | A Standard Concept representing the disease phase, outcome, or other abstraction of which the episode consists. For example, if the EPISODE\_CONCEPT\_ID is [treatment regimen](https://athena.ohdsi.org/search-terms/terms/32531) then the EPISODE\_OBJECT\_CONCEPT\_ID should contain the chemotherapy regimen concept, like [Afatinib monotherapy](https://athena.ohdsi.org/search-terms/terms/35804392). | Episode entries from the ‘Disease Episode’ concept class should have an episode\_object\_concept\_id that comes from the Condition domain. Episode entries from the ‘Treatment Episode’ concept class should have an episode\_object\_concept\_id that scome from the ‘Procedure’ domain or ‘Regimen’ concept class. | integer | Yes | No | Yes | CONCEPT | Procedure, Regimen |
| **episode\_type\_concept\_id** | This field can be used to determine the provenance of the Episode record, as in whether the episode was from an EHR system, insurance claim, registry, or other sources. | Choose the EPISODE\_TYPE\_CONCEPT\_ID that best represents the provenance of the record. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?domain=Type+Concept&standardConcept=Standard&page=1&pageSize=15&query=). A more detailed explanation of each Type Concept can be found on the [vocabulary wiki](https://github.com/OHDSI/Vocabulary-v5.0/wiki/Vocab.-TYPE_CONCEPT). | integer | Yes | No | Yes | CONCEPT | Type Concept |
| **episode\_source\_value** | The source code for the Episode 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. |  | varchar(50) | No | No | No |  |  |
| **episode\_source\_concept\_id** | A foreign key to a Episode Concept that refers to the code used in the source. | Given that the Episodes are user-defined it is unlikely that there will be a Source Concept available. If that is the case then set this field to zero. | integer | No | No | Yes | CONCEPT |  |

### episode\_event

**Table Description**

The EPISODE\_EVENT table connects qualifying clinical events (such as CONDITION\_OCCURRENCE, DRUG\_EXPOSURE, PROCEDURE\_OCCURRENCE, MEASUREMENT) to the appropriate EPISODE entry. For example, linking the precise location of the metastasis (cancer modifier in MEASUREMENT) to the disease episode.

**User Guide**

This connecting table is used instead of the FACT\_RELATIONSHIP table for linking low-level events to abstracted Episodes.

**ETL Conventions**

Some episodes may not have links to any underlying clinical events. For such episodes, the EPISODE\_EVENT table is not populated.

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **episode\_id** | Use this field to link the EPISODE\_EVENT record to its EPISODE. | Put the EPISODE\_ID that subsumes the EPISODE\_EVENT record here. | integer | Yes | No | Yes | EPISODE |  |
| **event\_id** | This field is the primary key of the linked record in the database. For example, if the Episode Event is a Condition Occurrence, then the CONDITION\_OCCURRENCE\_ID of the linked record goes in this field. | Put the primary key of the linked record here. | integer | Yes | No | No |  |  |
| **episode\_event\_field\_concept\_id** | This field is the CONCEPT\_ID that identifies which table the primary key of the linked record came from. | Put the CONCEPT\_ID that identifies which table and field the EVENT\_ID came from. [Accepted Concepts](https://athena.ohdsi.org/search-terms/terms?vocabulary=CDM&conceptClass=Field&page=1&pageSize=15&query=) | integer | Yes | No | Yes | CONCEPT | Metadata |

### metadata

**Table Description**

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

**User Guide**

NA

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **metadata\_id** | The unique key given to a Metadata record. | Attribute value is auto-generated | integer | Yes | Yes | No |  |  |
| **metadata\_concept\_id** |  |  | integer | Yes | No | Yes | CONCEPT |  |
| **metadata\_type\_concept\_id** |  |  | integer | Yes | No | Yes | CONCEPT |  |
| **name** |  |  | varchar(250) | Yes | No | No |  |  |
| **value\_as\_string** |  |  | varchar(250) | No | No | No |  |  |
| **value\_as\_concept\_id** |  |  | integer | No | No | Yes | CONCEPT |  |
| **value\_as\_number** | This is the numerical value of the result of the Metadata, if applicable and available. It is not expected that all Metadata will have numeric results, rather, this field is here to house values should they exist. |  | float | No | No | No |  |  |
| **metadata\_date** |  |  | date | No | No | No |  |  |
| **metadata\_datetime** |  |  | datetime | No | No | No |  |  |

### cdm\_source

**Table Description**

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

**User Guide**

NA

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **cdm\_source\_name** | The name of the CDM instance. |  | varchar(255) | Yes | No | No |  |  |
| **cdm\_source\_abbreviation** | The abbreviation of the CDM instance. |  | varchar(25) | Yes | No | No |  |  |
| **cdm\_holder** | The holder of the CDM instance. |  | varchar(255) | Yes | No | No |  |  |
| **source\_description** | The description of the CDM instance. |  | varchar(MAX) | No | No | No |  |  |
| **source\_documentation\_reference** |  |  | varchar(255) | No | No | No |  |  |
| **cdm\_etl\_reference** |  | Version of the ETL script used. e.g. link to the Git release | varchar(255) | No | No | No |  |  |
| **source\_release\_date** | The date the data was extracted from the source system. In some systems that is the same as the date the ETL was run. Typically the latest even date in the source is on the source\_release\_date. |  | date | Yes | No | No |  |  |
| **cdm\_release\_date** | The date the ETL script was completed. Typically this is after the source\_release\_date. |  | date | Yes | No | No |  |  |
| **cdm\_version** | Version of the OMOP CDM used as string. e.g. v5.4 |  | varchar(10) | No | No | No |  |  |
| **cdm\_version\_concept\_id** | The Concept Id representing the version of the CDM. | You can find all concepts that represent the CDM versions using the query: SELECT \* FROM CONCEPT WHERE VOCABULARY\_ID = 'CDM' AND CONCEPT\_CLASS = 'CDM' | integer | Yes | No | Yes | CONCEPT |  |
| **vocabulary\_version** | Version of the OMOP standardised vocabularies loaded | You can find the version of your Vocabulary using the query: SELECT vocabulary\_version from vocabulary where vocabulary\_id = 'None' | varchar(20) | Yes | No | No |  |  |

### concept

**Table Description**

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

**User Guide**

The primary purpose of the CONCEPT table is to provide a standardized representation of medical Concepts, allowing for consistent querying and analysis across the healthcare databases. Users can join the CONCEPT table with other tables in the CDM to enrich clinical data with standardized Concept information or use the CONCEPT table as a reference for mapping clinical data from source terminologies to Standard Concepts.

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **concept\_id** | A unique identifier for each Concept across all domains. |  | integer | Yes | Yes | No |  |  |
| **concept\_name** | An unambiguous, meaningful and descriptive name for the Concept. |  | varchar(255) | Yes | No | No |  |  |
| **domain\_id** | A foreign key to the [DOMAIN](https://ohdsi.github.io/CommonDataModel/cdm531.html#domain) table the Concept belongs to. |  | varchar(20) | Yes | No | Yes | DOMAIN |  |
| **vocabulary\_id** | A foreign key to the [VOCABULARY](https://ohdsi.github.io/CommonDataModel/cdm531.html#vocabulary) table indicating from which source the Concept has been adapted. |  | varchar(20) | Yes | No | Yes | VOCABULARY |  |
| **concept\_class\_id** | The attribute or concept class of the Concept. Examples are ‘Clinical Drug’, ‘Ingredient’, ‘Clinical Finding’ etc. |  | varchar(20) | Yes | No | Yes | CONCEPT\_CLASS |  |
| **standard\_concept** | This flag determines where a Concept is a Standard Concept, i.e. is used in the data, a Classification Concept, or a non-standard Source Concept. The allowable values are ‘S’ (Standard Concept) and ‘C’ (Classification Concept), otherwise the content is NULL. |  | varchar(1) | No | No | No |  |  |
| **concept\_code** | 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. |  | varchar(50) | Yes | No | No |  |  |
| **valid\_start\_date** | The date when the Concept was first recorded. The default value is 1-Jan-1970, meaning, the Concept has no (known) date of inception. |  | date | Yes | No | No |  |  |
| **valid\_end\_date** | The date when the Concept became invalid because it was deleted or superseded (updated) by a new concept. The default value is 31-Dec-2099, meaning, the Concept is valid until it becomes deprecated. |  | date | Yes | No | No |  |  |
| **invalid\_reason** | 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. |  | varchar(1) | No | No | No |  |  |

### vocabulary

**Table Description**

The VOCABULARY table includes a list of the Vocabularies integrated from various sources or created de novo in OMOP CDM. This reference table contains a single record for each Vocabulary and includes a descriptive name and other associated attributes for the Vocabulary.

**User Guide**

The primary purpose of the VOCABULARY table is to provide explicit information about specific vocabulary versions and the references to the sources from which they are asserted. Users can identify the version of a particular vocabulary used in the database, enabling consistency and reproducibility in data analysis. Besides, users can check the vocabulary release version in their CDM which refers to the vocabulary\_id = ‘None’.

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **vocabulary\_id** | A unique identifier for each Vocabulary, such as ICD9CM, SNOMED, Visit. |  | varchar(20) | Yes | Yes | No |  |  |
| **vocabulary\_name** | The name describing the vocabulary, for example, International Classification of Diseases, Ninth Revision, Clinical Modification, Volume 1 and 2 (NCHS) etc. |  | varchar(255) | Yes | No | No |  |  |
| **vocabulary\_reference** | External reference to documentation or available download of the about the vocabulary. |  | varchar(255) | No | No | No |  |  |
| **vocabulary\_version** | Version of the Vocabulary as indicated in the source. |  | varchar(255) | No | No | No |  |  |
| **vocabulary\_concept\_id** | A Concept that represents the Vocabulary the VOCABULARY record belongs to. |  | integer | Yes | No | Yes | CONCEPT |  |

### domain

**Table Description**

The DOMAIN table includes a list of OMOP-defined Domains to which the Concepts of the Standardized Vocabularies can belong. A Domain represents a clinical definition whereby we assign matching Concepts for the standardized fields in the CDM tables. For example, the Condition Domain contains Concepts that describe a patient condition, and these Concepts can only be used in the condition\_concept\_id field of the CONDITION\_OCCURRENCE and CONDITION\_ERA tables. This reference table is populated with a single record for each Domain, including a Domain ID and a descriptive name for every Domain.

**User Guide**

Users can leverage the DOMAIN table to explore the full spectrum of health-related data Domains available in the Standardized Vocabularies. Also, the information in the DOMAIN table may be used as a reference for mapping source data to OMOP domains, facilitating data harmonization and interoperability.

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **domain\_id** | A unique key for each domain. |  | varchar(20) | Yes | Yes | No |  |  |
| **domain\_name** | The name describing the Domain, e.g. Condition, Procedure, Measurement etc. |  | varchar(255) | Yes | No | No |  |  |
| **domain\_concept\_id** | A Concept representing the Domain Concept the DOMAIN record belongs to. |  | integer | Yes | No | Yes | CONCEPT |  |

### concept\_class

**Table Description**

The CONCEPT\_CLASS table includes semantic categories that reference the source structure of each Vocabulary. Concept Classes represent so-called horizontal (e.g. MedDRA, RxNorm) or vertical levels (e.g. SNOMED) of the vocabulary structure. Vocabularies without any Concept Classes, such as HCPCS, use the vocabulary\_id as the Concept Class. This reference table is populated with a single record for each Concept Class, which includes a Concept Class ID and a fully specified Concept Class name.

**User Guide**

Users can utilize the CONCEPT\_CLASS table to explore the different classes or categories of concepts within the OHDSI vocabularies.

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **concept\_class\_id** | A unique key for each class. |  | varchar(20) | Yes | Yes | No |  |  |
| **concept\_class\_name** | The name describing the Concept Class, e.g. Clinical Finding, Ingredient, etc. |  | varchar(255) | Yes | No | No |  |  |
| **concept\_class\_concept\_id** | A Concept that represents the Concept Class. |  | integer | Yes | No | Yes | CONCEPT |  |

### concept\_relationship

**Table Description**

The CONCEPT\_RELATIONSHIP table contains records that define relationships between any two Concepts and the nature or type of the relationship. This table captures various types of relationships, including hierarchical, associative, and other semantic connections, enabling comprehensive analysis and interpretation of clinical concepts. Every kind of relationship is defined in the RELATIONSHIP table.

**User Guide**

The CONCEPT\_RELATIONSHIP table can be used to explore hierarchical or attribute relationships between concepts to understand the hierarchical structure of clinical concepts and uncover implicit connections and associations within healthcare data. For example, users can utilize mapping relationships (‘Maps to’) to harmonize data from different sources and terminologies, enabling interoperability and data integration across disparate datasets.

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **concept\_id\_1** |  |  | integer | Yes | No | Yes | CONCEPT |  |
| **concept\_id\_2** |  |  | integer | Yes | No | Yes | CONCEPT |  |
| **relationship\_id** | The relationship between CONCEPT\_ID\_1 and CONCEPT\_ID\_2. Please see the [Vocabulary Conventions](https://ohdsi.github.io/CommonDataModel/dataModelConventions.html#concept_relationships). for more information. |  | varchar(20) | Yes | No | Yes | RELATIONSHIP |  |
| **valid\_start\_date** | The date when the relationship is first recorded. |  | date | Yes | No | No |  |  |
| **valid\_end\_date** | The date when the relationship is invalidated. |  | date | Yes | No | No |  |  |
| **invalid\_reason** | Reason the relationship was invalidated. Possible values are ‘D’ (deleted), ‘U’ (updated) or NULL. |  | varchar(1) | No | No | No |  |  |

### relationship

**Table Description**

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

**User Guide**

NA

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **relationship\_id** | The type of relationship captured by the relationship record. |  | varchar(20) | Yes | Yes | No |  |  |
| **relationship\_name** |  |  | varchar(255) | Yes | No | No |  |  |
| **is\_hierarchical** | Defines whether a relationship defines concepts into classes or hierarchies. Values are 1 for hierarchical relationship or 0 if not. |  | varchar(1) | Yes | No | No |  |  |
| **defines\_ancestry** | Defines whether a hierarchical relationship contributes to the concept\_ancestor table. These are subsets of the hierarchical relationships. Valid values are 1 or 0. |  | varchar(1) | Yes | No | No |  |  |
| **reverse\_relationship\_id** | The identifier for the relationship used to define the reverse relationship between two concepts. |  | varchar(20) | Yes | No | No |  |  |
| **relationship\_concept\_id** | A foreign key that refers to an identifier in the [CONCEPT](https://ohdsi.github.io/CommonDataModel/cdm531.html#concept) table for the unique relationship concept. |  | integer | Yes | No | Yes | CONCEPT |  |

### concept\_synonym

**Table Description**

The CONCEPT\_SYNONYM table is used to store alternate names and descriptions for Concepts.

**User Guide**

NA

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **concept\_id** |  |  | integer | Yes | No | Yes | CONCEPT |  |
| **concept\_synonym\_name** |  |  | varchar(1000) | Yes | No | No |  |  |
| **language\_concept\_id** |  |  | integer | Yes | No | Yes | CONCEPT |  |

### concept\_ancestor

**Table Description**

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

This table is entirely derived from the CONCEPT, CONCEPT\_RELATIONSHIP and RELATIONSHIP tables.

**User Guide**

NA

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ancestor\_concept\_id** | The Concept Id for the higher-level concept that forms the ancestor in the relationship. |  | integer | Yes | No | Yes | CONCEPT |  |
| **descendant\_concept\_id** | The Concept Id for the lower-level concept that forms the descendant in the relationship. |  | integer | Yes | No | Yes | CONCEPT |  |
| **min\_levels\_of\_separation** | 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. |  | integer | Yes | No | No |  |  |
| **max\_levels\_of\_separation** | 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. |  | integer | Yes | No | No |  |  |

### source\_to\_concept\_map

**Table Description**

The source to concept map table is 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. **There are OHDSI tools to help you populate this table;** [**Usagi**](https://github.com/OHDSI/Usagi) **and** [**Perseus**](https://github.com/ohdsi/Perseus)**. You can read more about OMOP vocabulary mapping in** [**The Book of OHDSI Chapter 6.3**](https://ohdsi.github.io/TheBookOfOhdsi/ExtractTransformLoad.html#step-2-create-the-code-mappings)**.**

**User Guide**

NA

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **source\_code** | The source code being translated into a Standard Concept. |  | varchar(50) | Yes | No | No |  |  |
| **source\_concept\_id** | A foreign key to the Source Concept that is being translated into a Standard Concept. | This is either 0 or should be a number above 2 billion, which are the Concepts reserved for site-specific codes and mappings. | integer | Yes | No | Yes | CONCEPT |  |
| **source\_vocabulary\_id** | A foreign key to the VOCABULARY table defining the vocabulary of the source code that is being translated to a Standard Concept. |  | varchar(20) | Yes | No | No |  |  |
| **source\_code\_description** | 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. |  | varchar(255) | No | No | No |  |  |
| **target\_concept\_id** | The target Concept to which the source code is being mapped. |  | integer | Yes | No | Yes | CONCEPT |  |
| **target\_vocabulary\_id** | The Vocabulary of the target Concept. |  | varchar(20) | Yes | No | Yes | VOCABULARY |  |
| **valid\_start\_date** | The date when the mapping instance was first recorded. |  | date | Yes | No | No |  |  |
| **valid\_end\_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. |  | date | Yes | No | No |  |  |
| **invalid\_reason** | 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. |  | varchar(1) | No | No | No |  |  |

### drug\_strength

**Table Description**

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

**User Guide**

NA

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **drug\_concept\_id** | The Concept representing the Branded Drug or Clinical Drug Product. |  | integer | Yes | No | Yes | CONCEPT |  |
| **ingredient\_concept\_id** | The Concept representing the active ingredient contained within the drug product. | Combination Drugs will have more than one record in this table, one for each active Ingredient. | integer | Yes | No | Yes | CONCEPT |  |
| **amount\_value** | The numeric value or the amount of active ingredient contained within the drug product. |  | float | No | No | No |  |  |
| **amount\_unit\_concept\_id** | The Concept representing the Unit of measure for the amount of active ingredient contained within the drug product. |  | integer | No | No | Yes | CONCEPT |  |
| **numerator\_value** | The concentration of the active ingredient contained within the drug product. |  | float | No | No | No |  |  |
| **numerator\_unit\_concept\_id** | The Concept representing the Unit of measure for the concentration of active ingredient. |  | integer | No | No | Yes | CONCEPT |  |
| **denominator\_value** | The amount of total liquid (or other divisible product, such as ointment, gel, spray, etc.). |  | float | No | No | No |  |  |
| **denominator\_unit\_concept\_id** | The Concept representing the denominator unit for the concentration of active ingredient. |  | integer | No | No | Yes | CONCEPT |  |
| **box\_size** | The number of units of Clinical Branded Drug or Quantified Clinical or Branded Drug contained in a box as dispensed to the patient. |  | integer | No | No | No |  |  |
| **valid\_start\_date** | The date when the Concept was first recorded. The default value is 1-Jan-1970. |  | date | Yes | No | No |  |  |
| **valid\_end\_date** | The date when then Concept became invalid. |  | date | Yes | No | No |  |  |
| **invalid\_reason** | 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. |  | varchar(1) | No | No | No |  |  |

### cohort

**Table Description**

The subject of a cohort can have multiple, discrete records in the cohort table per cohort\_definition\_id, subject\_id, and non-overlapping time periods. The definition of the cohort is contained within the COHORT\_DEFINITION table. It is listed as part of the RESULTS schema because it is a table that users of the database as well as tools such as ATLAS need to be able to write to. The CDM and Vocabulary tables are all read-only so it is suggested that the COHORT and COHORT\_DEFINTION tables are kept in a separate schema to alleviate confusion.

**User Guide**

NA

**ETL Conventions**

Cohorts typically include patients diagnosed with a specific condition, patients exposed to a particular drug, but can also be Providers who have performed a specific Procedure. Cohort records must have a Start Date and an End Date, but the End Date may be set to Start Date or could have an applied censor date using the Observation Period Start Date. Cohort records must contain a Subject Id, which can refer to the Person, Provider, Visit record or Care Site though they are most often Person Ids. The Cohort Definition will define the type of subject through the subject concept id. A subject can belong (or not belong) to a cohort at any moment in time. A subject can only have one record in the cohort table for any moment of time, i.e. it is not possible for a person to contain multiple records indicating cohort membership that are overlapping in time

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **cohort\_definition\_id** |  |  | integer | Yes | No | No |  |  |
| **subject\_id** |  |  | integer | Yes | No | No |  |  |
| **cohort\_start\_date** |  |  | date | Yes | No | No |  |  |
| **cohort\_end\_date** |  |  | date | Yes | No | No |  |  |

### cohort\_definition

**Table Description**

The COHORT\_DEFINITION table contains records defining a Cohort derived from the data through the associated description and syntax and upon instantiation (execution of the algorithm) placed into the COHORT table. Cohorts are a set of subjects that satisfy a given combination 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.

**User Guide**

NA

**ETL Conventions**

NA

| **CDM Field** | **User Guide** | **ETL Conventions** | **Datatype** | **Required** | **Primary Key** | **Foreign Key** | **FK Table** | **FK Domain** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **cohort\_definition\_id** | This is the identifier given to the cohort, usually by the ATLAS application |  | integer | Yes | No | No |  |  |
| **cohort\_definition\_name** | A short description of the cohort |  | varchar(255) | Yes | No | No |  |  |
| **cohort\_definition\_description** | A complete description of the cohort. |  | varchar(MAX) | No | No | No |  |  |
| **definition\_type\_concept\_id** | Type defining what kind of Cohort Definition the record represents and how the syntax may be executed. |  | integer | Yes | No | Yes | CONCEPT |  |
| **cohort\_definition\_syntax** | Syntax or code to operationalize the Cohort Definition. |  | varchar(MAX) | No | No | No |  |  |
| **subject\_concept\_id** | This field contains a Concept that represents the domain of the subjects that are members of the cohort (e.g., Person, Provider, Visit). |  | integer | Yes | No | Yes | CONCEPT |  |
| **cohort\_initiation\_date** | A date to indicate when the Cohort was initiated in the COHORT table. |  | date | No | No | No |  |  |