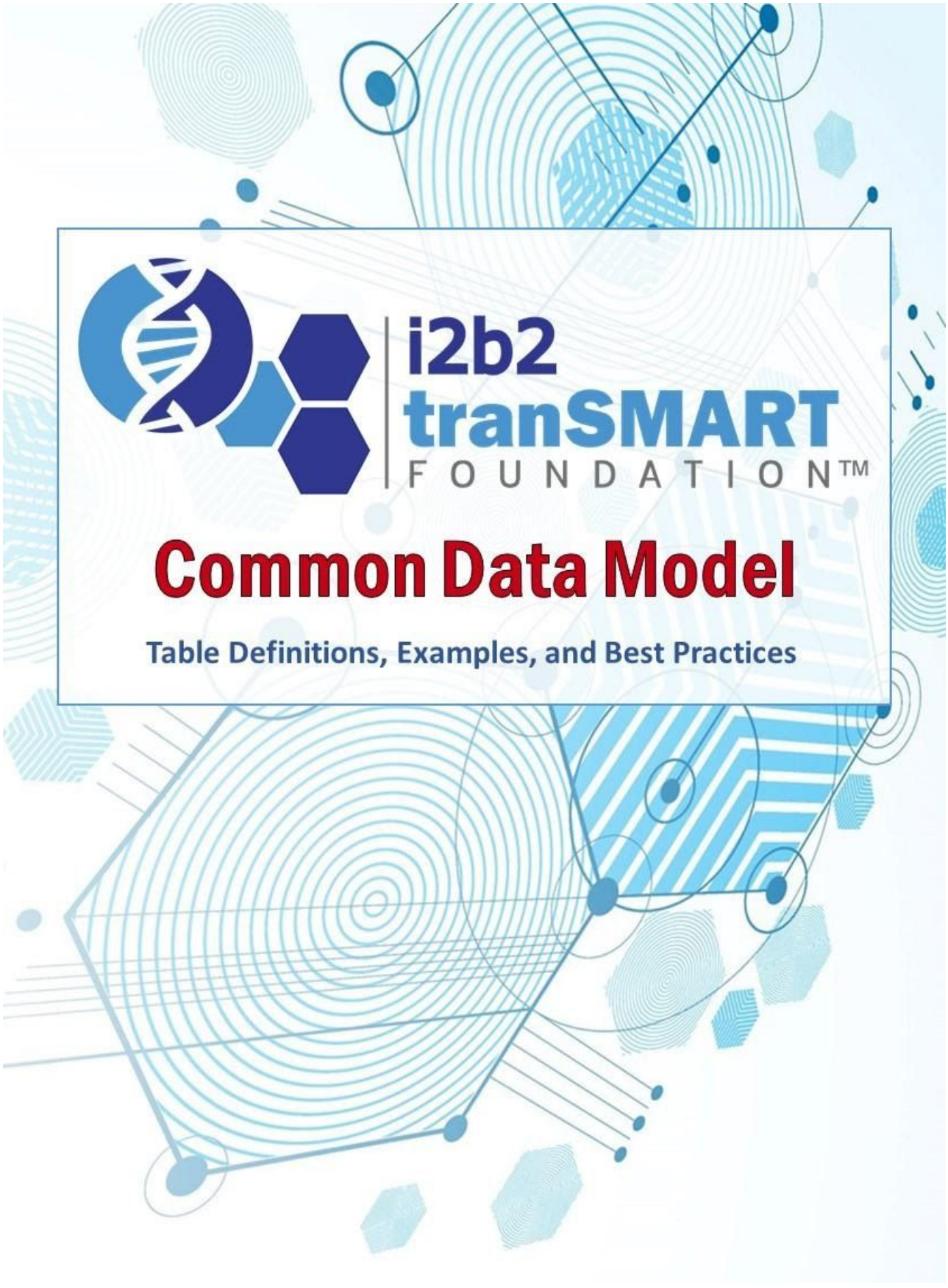
i2b2
tranSMART
FOUNDATION™
Common Data Model
Table Definitions, Examples, and Best Practices

Version History

Version #	Release Date	Description
v0.9	11-06-2020	Pre-release for community feedback

Contents

1. Introduction

i2b2 and tranSMART were developed to provide clinical and translational investigators with the tools necessary to integrate medical record and clinical research data in the genomics age. The core of this is a highly flexible but simple i2b2 Common Data Model (CDM). The current version of the i2b2 CDM was released in July, 2020. This document not only describes the database tables and fields in the i2b2 CDM, but also provides a set of recommendations and best practices for using it.

The i2b2 CDM, which we initially developed in 2004, is based on a "star schema". Instead of separate tables for diagnoses, medications, and other data types, all patient observations are stored in a single "fact" table. A separate ontology describes the different codes that are placed in this fact table. As a result, institutions can use their own local codes, without having to map to common code sets. Furthermore, institutions can easily add new types of data to i2b2 and tranSMART just by extending the ontology. No changes to the database or software are needed. This enables software developers to build query, analysis, and visualization tools that are generalizable to different types of data and future-proof since the i2b2 CDM can remain stable over time.

Over 200 institutions worldwide use the i2b2 CDM to store and integrate coded electronic health record and medical claims data, notes, images, genomics, clinical trial data and more. It is highly scalable, with some instances containing billions of data facts for millions of patients and supporting queries from thousands of users. It is open source and has been implemented for Microsoft SQL Server, Oracle, and Postgres. The i2b2 CDM databases at different institutions can be linked and harmonized to form large federated data networks using the Shared Health Research Information Network (SHRINE) software. Because it is ontology based, the i2b2 CDM is well suited to address rapidly emerging public health crises, such as COVID-19, since codes for new tests and diagnoses only require updates to the ontology, not the database.

Other popular data models, such as OMOP CDM, have separate database tables for each data type. Although their schemas are more complex, they can be easier to learn for people who have not used ontology-based systems. (Nevertheless, the numerous benefits of the i2b2 CDM has led to its widespread adoption.) Aware of this, we have designed this documentation to start with the basics in a Quick Start guide to help new institutions gain familiarity with the i2b2 CDM (Chapter 2). Next, we provide a schema reference, introducing each of the tables and fields in the i2b2 CDM (Chapter 3). Then, a series of brief tutorials explain more complex ways of using the i2b2 CDM to model different data types (Chapter 4). Finally, we discuss advanced topics, such as security and performance optimization (Chapter 5).

2. Quick Start Guide

This chapter provides a brief overview of the i2b2 CDM star schema and how to use it in combination with an ontology.

2.1. Star Schema Introduction

In this section we describe, at a high level, the main tables and fields in the i2b2 CDM. A detailed description of the full data model is in the i2b2 CDM spreadsheet.

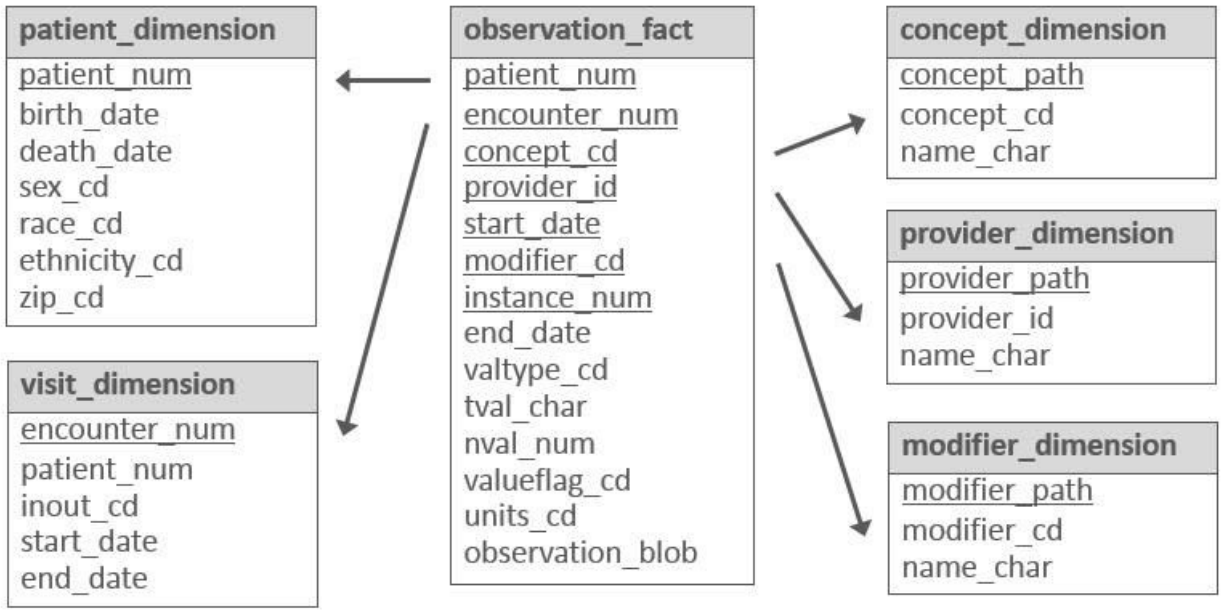

Figure 1. Main tables and fields in the i2b2 CDM star schema. Underlined fields are part of the primary keys of tables. Note that not all tables and fields in the i2b2 CDM are shown here. See the i2b2 CDM spreadsheet for the full data model.

The i2b2 CDM is a data warehouse modeled on the star schema structure first proposed by Ralph Kimball. The database schema looks like a star, with one central fact table surrounded by one or more dimension tables. The most important concept regarding the construction of a star schema is identifying what constitutes a fact.

In healthcare, a logical fact is an observation on a patient. It is important to note that an observation may not represent the onset or date of the condition or event being described, but instead is simply a

recording or a notation of something. For example, the observation of 'diabetes' recorded in the database as a 'fact' at a particular time does not mean that the condition of diabetes began exactly at that time, only that a diagnosis was recorded at that time (there may be many diagnoses of diabetes for this patient over time).

The fact table contains the basic attributes about the observation, such as the patient and provider numbers, a concept code for the concept observed, a start and end date, and other parameters described in this document. In the i2b2, the fact table is **OBSERVATION_FACT**. All patient observations are placed in this table, such as diagnoses, procedures, medications, and laboratory test results. A large institution with millions of patients might have billions of rows of observations in this table.

Dimension tables contain further descriptive and analytical information about attributes in the fact table. A dimension table may contain information about how certain data is organized, such as a hierarchy that can be used to categorize or summarize the data. In the i2b2 data mart, there are five dimension tables that provide additional information about fields in the fact table: **PATIENT_DIMENSION**, **CONCEPT_DIMENSION**, **VISIT_DIMENSION**, **PROVIDER_DIMENSION**, **MODIFIER_DIMENSION**.

2.2. Table Descriptions and Examples

In the **OBSERVATION_FACT** table, observation codes, such as ICD-10 diagnosis or NDC medication codes, are placed in the concept_cd field. In addition to the concept_cd, the patient_num, encounter_num, and start_date fields are required for each observation. A provider (observer) ID, modifier code and instance number are used for certain types of observations. An “@” symbol is used as a default value for the provider_id and modifier_cd, and 1 is the default instance_num.

The fact table also contains value objects associated with the observations. A laboratory test result, with a single value, can be stored in one row of the fact table. A complex value, such as blood pressure, for example “120/80-standing”, is reduced to three rows with concept “blood pressure” and modifiers “systolic”, “diastolic”, and “position” (preferably expressed as LOINC or other known standard, but this is not required). The value type of the observation is specified in the valtype_cd field, such as N=number, T=text, D=date, and so forth. Numbers and dates are placed in the nval_char field which is a numeric data type. Text-based values are placed in the tval_char field; and, binary data are stored in the observation_blob field. For numeric values, the tval_char field can be used to indicate an operator, such as “E” (equals), “G” (greater than), or “LE” (less than or equal to). For example, tval_char=“L” and nval_num=“0.01” means the value is less than 0.01.

patient_num	encounter_num	concept_cd	start_date	modifier_cd	valtype_cd	tval_char	nval_num
1000001	123456	ICD9:462	2007-08-09	@			
1000001	123456	LOINC:1751-7	2007-08-10	@	N	E	4.3
1000001	298765	LOINC:6598-7	2007-09-15	@	N	L	0.01
1000002	890123	LOINC:6598-7	2009-03-20	@	T	NEG	
1000002	543210	VITAL:BP	2010-05-01	systolic	N	120	

1000002	543210	VITAL:BP	2010-05-01	diastolic	N	80	
1000002	543210	VITAL:BP	2010-05-01	position	T	standing	

Table 1. Example observations in the fact table. Shown are a basic diagnosis fact (ICD9 code), laboratory tests (LOINC codes) with numeric and text based results, and a multi-part blood pressure observation stored as three facts (custom VITAL:BP code).

In addition to the fact table, there are five other dimension tables that help express the patient data. The **PATIENT_DIMENSION** table has one row for every patient in the database. The patient_num field is a unique integer for each patient. A separate **PATIENT_MAPPING** table optionally maps the patient_num to a medical record number or other local identifier, which may be non-numeric. The patient_dimension table contains several optional demographic fields, including birth_date, death_date, sex_cd, and race_cd. Demographic concepts can alternatively be placed as observations in the fact table, depending on how an institution choses to model the data.

The **VISIT_DIMENSION** table allows periods to be represented that correspond roughly to patient encounters where observations were recorded. An “encounter” can involve a patient directly, such as a visit to a doctor's office, or it can involve the patient indirectly, such as running several tests tied together by the same tube of the patient's blood. Similar to patient_num, the ecounter_num is a unique integer for each row in the visit_dimension table. A separate **ENCOUNTER_MAPPING** table optionally maps the encounter_num to local encounter billing codes, visit IDs, or other local identifier.

The **CONCEPT_DIMENSION** table has vocabulary terms that map to the codes used in the concept_cd field of the fact table. These terms typically come from standard terminologies, such as International Classification of Diseases (ICD), National Drug Code (NDC), and Logical Observation Identifiers Names and Codes (LOINC). However, the i2b2 and tranSMART software do not recognize a difference between standard and local terminologies. Terms may be grouped into hierarchies. The hierarchical representation used in the concept table is similar to that of a hierarchical file system. The parent term is positioned in the “folder” position of the path, and the child term in the “file” position. For example, in the concept_dimension table (Table 2), the parent “anti-infectives” can have the three children “penicillin”, “ampicillin”, and “Bactrim”. The children map to the NDC codes used in the fact table, but the path shows they are types of anti-infectives.

concept_path	**concept_cd**	**name_char**
\Med\		Medications
\Med\anti-infectives\		Anti-Infectives
\Med\anti-infectives\penicillin\	NDC:00002032902	Penicillin
\Med\anti-infectives\ampicillin\	NDC:60429002340	Ampicillin
\Med\anti-infectives\Bactrim\	NDC:00003013850	Bactrim

Table 2. Layout of concept data representations in the concept dimension. Paths group codes into a hierarchy, like a computer file system.

This hierarchical organization allows users to query for a path, such as the general concept of anti-infectives, and the i2b2 or tranSMART software can automatically and efficiently convert this to a set of concept codes to search for in the fact table. For example, the query below finds all patients seen on anti-infectives, we would run the following query:

```
Select distinct(patient_num)
From observation_fact
Where concept_cd in
     (select concept_cd
     from concept_dimension
     where concept_path like
     '\Med\anti-infectives\%')
```

The path of the concept is used to find and use all concept_cds that fall into the anti-infectives group. If we only wanted to find patients specifically on Bactrim, we would use the same query with the following concept_path: "\Med \anti-infectives\Bactrim\%".

The same approach is used in the **PROVIDER_DIMENSION** and **MODIFIER_DIMENSION** tables. A path can represent a group of providers, such as a hospital department, or the individual clinicians that are part of that department. Examples of modifiers include primary or secondary indicators for diagnoses and dose, route, and frequency for medications.

3. Table and Field Definitions and Descriptions

This chapter contains a complete reference list of tables and fields in the i2b2 CDM, followed by a detailed description of each of the tables.

3.1 Schema Reference

A spreadsheet describing all the tables and fields in the i2b2 CDM can be found at

https://docs.google.com/spreadsheets/d/1ZRX6JUhS-uC5Lz5rgBGi0vEJp99gGjDLLMH7Dnz0FLg

It contains a row for each table and field. The columns "**Data Type: MSSQL**", "**Data Type: Oracle**", and "**Data Type: Postgres**" indicate the data type of the field in Microsoft SQL Server (MSSQL), Oracle, and Postgres, respectively. The "**Primary Key**" column indicates which fields are primary or foreign keys.

The "**Core**" column indicates the fields that are currently used by the i2b2 and tranSMART software and required to be available in any implementation of the i2b2 CDM. The "**Admin**" column indicates administrative fields that are used to indicate the source system for the data, when data were imported into the table, when the data were last updated, etc. These fields are not used by query or analysis features within the i2b2 or tranSMART software. However, some ETL tools for i2b2 and tranSMART use them. The "**Future**" column indicates fields that are intended to be used in future versions of i2b2 and tranSMART. Fields that are neither "Core", "Admin", nor "Future" are optional. They represent common types of data. For example, the "sex_cd" and "race_cd" fields in the PATIENT_DIMENSION table are often used to store the sex and race of the patient. However, these fields are not required, and the software does not specify what codes to use. The customizable ontology in i2b2 and tranSMART defines whether these fields are used, what codes are allowed, and what those codes mean.

Additional fields can be added to any of the tables in the i2b2 CDM. The ontology defines how they are used. Any field that is not a Core field can be removed. For large amounts of data (e.g., billions of observations), removing unused Core fields can save significant disk space. Additional dimension tables can also be created. Again, the ontology defines how these dimension tables link back to the OBSERVATION_FACT table.

The "**Values**" column indicates fields for which certain values are required. The "**Description**" column contains a brief description of the field. The "**History**" column indicates the first version of i2b2 that contained the column. Note that for backwards compatibility, no fields have ever been removed from i2b2. The "**Notes**" column contains additional information about certain fields.

3.2 Table and Field Descriptions

Each of the i2b2 CDM tables have a set of **administrative columns**, which are primarily used by ETL processes. A description of these fields are at

https://community.i2b2.org/wiki/display/ServerSideDesign/General+Information

Detailed descriptions of the **OBSERVATION_FACT** table is at

https://community.i2b2.org/wiki/display/ServerSideDesign/OBSERVATION_FACT+Table

Descriptions of the **five dimension tables** are at

https://community.i2b2.org/wiki/display/ServerSideDesign/PATIENT_DIMENSION+Table

https://community.i2b2.org/wiki/display/ServerSideDesign/VISIT_DIMENSION+Table

https://community.i2b2.org/wiki/display/ServerSideDesign/PROVIDER_DIMENSION+Table

https://community.i2b2.org/wiki/display/ServerSideDesign/CONCEPT_DIMENSION+Table

https://community.i2b2.org/wiki/display/ServerSideDesign/MODIFIER_DIMENSION+Table

The **relationships** between these dimension tables and the fact table are described at

https://community.i2b2.org/wiki/display/ServerSideDesign/Joining+Columns

The i2b2 software uses two optional mapping tables, **PATIENT_MAPPING** and **ENCOUNTER_MAPPING**, which map the i2b2 patient_num and encounter_num to local codes, such as medical record numbers or encounter billing IDs. These tables are not part of the core i2b2 CDM, but they are used by many sites. Descriptions of these tables are at

https://community.i2b2.org/wiki/display/ServerSideDesign/PATIENT_MAPPING+Table

https://community.i2b2.org/wiki/display/ServerSideDesign/ENCOUNTER_MAPPING+Table

4. Tutorials: Using the i2b2 CDM

This chapter contains a series of tutorials that describe how to model different types of data using the i2b2 CDM. The tutorials start with simpler, more common use cases, and progress to more advanced concepts.

4.1 Diagnoses and Demographics (Basic Facts)

Below are examples of basic facts in the i2b2 CDM.

A diagnosis, such as acute pharyngitis, has an associated code, such as "ICD10:J02.9". The customizable i2b2 ontology defines the concept and codes. Different sites might use their own codes for the acute pharyngitis. The observation of this diagnosis is stored as a record in the OBSERVATION_FACT table. A patient_num (1000001), encounter_num (730868), and start_date (2017-10-22) are required. A provider_id, modifier_cd, and instance_num are also required, but these can be set to their default values, "@", "@" and 1.

The patient_num and encounter_num corresponding to these diagnoses must be listed in the PATIENT_DIMENSION and VISIT_DIMENSION tables. In this example, the patient is female (sex_cd=F) and white (race_cd=W). The encounter is an outpatient visit (inout_cd=O), with start_date and end_date both October 22, 2017.

Note that the date of the observation in the OBSERVATION_FACT table can be different from the dates in the VISIT_DIMENSION. This is illustrated in a second diagnosis of asthma (ICD10:J45.909), which was recorded the day after the patient was discharged from an inpatient visit (encounter_num=798502). This diagnosis also has a provider (the observer). The PROVIDER_DIMENSION lists the provider (provider_id=X1824); it indicates her name is Jane Smith; and, the path can be used to group providers, for example, by department or specialty. In this example, the asthma diagnosis is duplicated. When this occurs, the instance_num must be incremented to ensure that the primary keys of each record are unique.

The i2b2 ontology enables concepts to be modeled in different ways. For example, the ontology can specify that the race_cd field of the PATIENT_DIMENSION be used to store the patient's race. Having all demographics in the PATIENT_DIMENSION table can simplify certain analyses. Alternatively, race can be an observation stored in the OBSERVATION_FACT table (e.g., concept_cd="DEM|Race:W"). This can be useful if demographics are separately recorded for each admission, or if there is a need to store multiple races for a patient.

The CONCEPT_DIMENSION table has vocabulary terms that map to the codes used in the concept_cd field of the OBSERVATION_FACT table. Terms may be grouped into hierarchies. The hierarchical representation used in the concept table is similar to that of a hierarchical file system. The parent term

is positioned in the "folder" position of the path, and the child term in the "file" position. In the examples here, acute pharyngitis and asthma both have paths that are children of the term "Diseases of the Respiratory System". In the i2b2 and tranSMART software, a query for the parent term will match the concept_cd codes corresponding to any of the child nodes.

OBSERVATION_FACT

patient_num	encounter_num	concept_cd	provider_id	start_date	modifier_cd	instance_num
1000001	730868	ICD10:J02.9	@	2017-10-22	@	1
1000001	798502	ICD10:J45.909	X1824	2018-02-12	@	1
1000001	798502	ICD10:J45.909	X1824	2018-02-12	@	2
1000001	798502	DEM|Race:W	@	2018-02-08	@	1

PATIENT_DIMENSION

patient_num	sex_cd	race_cd
1000001	F	W

VISIT_DIMENSION

encounter_num	patient_num	inout_cd	start_date	end_date
730868	1000001	O	2017-10-22	2017-10-22
798502	1000001	I	2018-02-08	2018-02-11

PROVIDER_DIMENSION

provider_path	provider_id	name_char
Medicine\Pulmonary\X1824\	X1824	Jane Smith, MD

CONCEPT_DIMENSION

concept_path	concept_cd	name_char
Dem\Race\White\	DEM|Race:W	White
Diag\ICD10\J00-J99\	ICD10:J	Diseases of the Respiratory System
Diag\ICD10\J00-J99\J02\J02.9\	ICD10:J02.9	Acute pharyngitis, unspecified
Diag\ICD10\J00-J99\J45\J45.909\	ICD10:J45.909	Unspecified asthma, uncomplicated

Figure 2. Example data in the i2b2 CDM.

4.2 Laboratory Tests and Vital Signs (Value Constraints)

Laboratory tests, vital signs, and other data types can have an associated value. The OBSERVATION_FACT table contains six fields to store values: VALTYPE_CD, TVAL_CHAR, NVAL_CHAR, VALUEFLAG_CD, UNITS_CD, and OBSERVATION_BLOB. A description of these fields are at

https://community.i2b2.org/wiki/display/ServerSideDesign/Value+Columns

Note that the i2b2 and tranSMART ontology must be configured properly so that the software uses these value-related fields. A description of how the ontology works with the value fields is at

https://community.i2b2.org/wiki/display/ServerSideDesign/Example+of+Value+Constraints+Used+in+Queries

Additional information about the i2b2 Ontology Management Cell can be found at

https://community.i2b2.org/wiki/display/ServerSideDesign/Ontology+Management+%28ONT%29+Cell

4.3 Medications, Procedures and Allergies (Modifiers)

A single observation can be represented in the i2b2 CDM as a fact with an unlimited number of modifier codes. A medication, for example, can be modified with dose, route, and frequency. Each of these modifiers is stored as a separate record in the OBSERVATION_FACT table.

For example, a prescription for Aspirin 325 mg QD PO on 4/4/2010 is stored as four records. All four have "med:aspirin" as the CONCEPT_CD and "4/4/2010" as the START_DATE. The "base" record for the observation uses "@" for the MODIFIER_CD. The three modifier records have MODIFIER_CD values of "MED:DOSE", "MED:FREQ", and "MED:ROUTE", with their associated values (325 mg, "QD", and "PO") stored in the value fields. In this example, the concept med:aspirin is defined in the CONCEPT_DIMENSION tables; and, the three modifier codes, MED:DOSE, MED:FREQ, and MED:ROUTE, are defined in the MODIFIER_DIMENSION table. The i2b2 and tranSMART software also require these to be included in the ontology. A more detailed description of this example and an example using procedures are available at

https://community.i2b2.org/wiki/display/DevForum/Modifiers+in+i2b2+Data+Model

Modifiers can be used for many other things, such as indicating whether a diagnosis is primary or secondary or adding stage to a cancer diagnosis. Another example using modifiers to indicate patients' allergies is at

https://community.i2b2.org/wiki/display/DevForum/Representing+Allergies+in+Star+Schema+with+modifiers

4.4 Clinical Notes, Imaging and Genomics Data (Blob Data)

Clinical Notes

There is a wealth of information within the plain text clinical narrative. Notes can be represented in the i2b2 CDM in a couple ways. First, the entire note can be a single record in the OBSERVATION_FACT table. The concept_cd can be, for example, "NOTE:DischargeSummary"; and, the text of the note is stored in the observation_blob field. This field contains a FULLTEXT index in the Microsoft SQL Server implementation of the i2b2 CDM to enable efficient searches of notes. A description of text search in i2b2 is at

https://community.i2b2.org/wiki/display/DevForum/Text+search+in+i2b2

Alternatively, natural language processing (NLP) software can be used to extract concepts from notes. Each concept can be stored as a separate record in the OBSERVATION_FACT table. An example of this using the NLP CTAKES program is at

https://community.i2b2.org/wiki/display/NLPCTAKES/NLP+cTakes+Home

Imaging Data

The observation_blob field can also be used to store binary data, such as medical images. An example of this was a project called mi2b2 (Medical Imaging Informatics Bench to Bedside), which linked i2b2 to separate PACS (Picture Archiving and Communication System) systems. Although the original images were stored in the PACS systems, a thumbnail version was placed in the i2b2 observation_blob field, so that it could be previewed within the i2b2 user interface. A description of mi2b2 is at

https://community.i2b2.org/wiki/display/mi2b2/mi2b2+User+Documentation

Genomics Data

Like notes, genomics data can be modeled in different ways. A tutorial showing how to load genomic VCF (Variant Call Format) files into the observation_blob field is at

https://community.i2b2.org/wiki/display/IGD/Demo+Data

4.5 Lab Panels and Specimens (Dummy Records)

The encounter_num typically represents a visit. However, it can also be used more generally to group related observations. A “dummy” encounter_num value, for example, can be created that corresponds to a lab panel, such as a complete blood count (CBC). The start_date for this encounter in the VISIT_DIMENSION table could be the specimen date. The results of each test within the panel are stored as separate records in the OBSERVATION_FACT table, using the same encounter_num.

4.6 Clinical Trial Data (Extending the Data Model)

Encounters in clinical trials are often associated with a visit number. The VISIT_DIMENSION table can be extended with a visit_num field to store the visit number. The ontology can include a concept for the visit number that points to this field, so that the user can query for observations that occurred as part of a particular visit number.

5. Advanced Topics

5.1 Encrypted Data

Data in the observation_blob field of the OBSERVATION_FACT table can be encrypted.

5.2 Multiple Fact Tables

Starting with release 1.7.09, the i2b2 software supports multiple fact tables. This enables, for example, diagnoses to be stored in one fact table, laboratory tests in a second fact table, and clinical notes in a third. This might be helpful to simplify data updates or for performance reasons in large databases. Details of how to configure i2b2 to use multiple fact tables is at

https://community.i2b2.org/wiki/display/MFT/Multi-fact+Table+Home?preview=%2F339480%2F339493%2Fmultifact-setup-guide.pdf

5.3 Working with OMOP Data

The Observational Medical Outcomes Partnership (OMOP) CDM is another widely adopted data model. Rather than a central fact table, the OMOP CDM uses separate tables for each data domain: procedures, condition, drug, measurement, observation, etc. The multiple fact table feature in i2b2 can be used to treat each OMOP table as a separate fact table, enabling i2b2 to run the OMOP CDM with the appropriate i2b2 ontology configuration. A description of this is at

https://community.i2b2.org/wiki/display/OMOP/OMOP+Home

5.4 Loading Data into the i2b2 CDM

The i2b2 tranSMART Foundation ETL Working Group has assembled a set of resources, including documentation and software, to assist in loading data into i2b2, which is at

https://community.i2b2.org/wiki/display/IWG/ETL+Working+Group

6. Acknowledgements

This document was written by Griffin M Weber, Jeff Klann, Mike Mendis, Shawn Murphy, Rudy Potenzone, and Peter Rice.