Title: The Generalized Data Model for Clinical Research

Authors: Mark D. Danese, MHS, PhD; Marc Halperin; Jennifer Duryea, MPH; Ryan Duryea

Affiliations:

Outcomes Insights, Inc., Westlake Village, CA

Key words:

Data model, claims data, electronic health records

Contact information:

Mark Danese, MHS, PhD

2801 Townsgate Road, Suite 330

Westlake Village, CA 91361

805-498-0034 (office)

805-715-8106 (fax)

Word Count:

4,784 (excluding abstract, acknowledgements, tables, and references)

1. Abstract (295 words)

1.1 Background

Most healthcare data sources store information within their own unique schemas, making reliable and reproducible research challenging. Consequently, researchers have adopted various data models to improve the efficiency of research. Transforming and loading data into these models is a labor-intensive process that can alter the original semantics of the original data. Therefore, we created a data model with a hierarchical structure that simplifies the transformation process and minimizes data alteration.

1.2 Methods

There were two design goals in constructing the tables and table relationships for the Generalized Data Model (GDM). The first was to use a single table for storing clinical codes in their original vocabularies to retain the original semantic representation of the data. The second was to retain hierarchical information present in the original data while retaining provenance. The model was tested by transforming synthetic Medicare data; Surveillance, Epidemiology, and End Results data linked to Medicare claims; and electronic health records from the Clinical Practice Research Datalink. We also tested a subsequent transformation from the GDM into the Sentinel data model.

1.3 Results

The resulting data model contains 19 tables, with the Clinical Codes, Contexts, and Collections tables serving as the core of the model, and containing most of the clinical, provenance, and hierarchical information. In addition, a Mapping table allows users to apply an arbitrarily complex set of relationships among vocabulary elements to facilitate automated analyses.

1.4 Conclusions

The GDM offers researchers a simple process for loading data, clear data provenance, and a clear path for users to transform their data into other data models. The GDM is designed to retain hierarchical relationships among data elements as well as the original semantic representation of the data, ensuring consistency in study building as part of a complete data pipeline for researchers.

2. Background and Significance

Healthcare data contains useful information for clinical researchers across a wide range of disciplines, including pharmacovigilance, epidemiology, and health services research. However, most data sources store information within their own unique schemas, making it difficult to develop software tools that ensure reliable and reproducible research. One solution to this problem is to create data models that standardize the storage of both the data and the relationships among data elements.[1]

In healthcare, several commonly used data models include those supported by the following organizations: Informatics for Integrating Biology and the Bedside (i2b2) [2–4], Observational Health Data Sciences and Informatics (OHDSI, managing the OMOP data model) [5–7], Sentinel [8–10], and PCORnet [11, 12], among others. The first, and biggest, challenge with any data model is the process of migrating the raw (source) data into the data model, referred to as the “extract, transform, and load” (ETL) process. The ETL process is particularly burdensome when one has to support multiple, large data sources, and to update them regularly.

The biggest difficulty with ETL is not the process of standardizing variable names and reorganizing the data; rather, the difficulty is standardizing the relationships among data elements. Since different healthcare data sources encode relationships in different ways, the ETL process can lose information, or create inaccurate information. The best example is creating a visit, a construct which, in most data models, is used to link information (e.g., diagnoses and procedures). Visits are challenging because administrative claims allow facilities and practitioners to invoice separately for their portions of the same medical encounter, and practitioners to bill for multiple interactions on a single invoice.[13] Even constructing visits from electronic medical records can be challenging when each interaction with a different provider (e.g., nurse, physician, pharmacist, etc.) is recorded separately.

To minimize the need to encode specific relationships that may not exist in the source data, we created a data model with a hierarchical structure that simplifies the ETL process without changing the meaning of the original data. This data model can serve both as a stand-alone data model for clinical researchers using observational data, as well as a common storage model for later conversion into other data models.

3. Materials and Methods

In designing the Generalized Data Model (GDM) the primary use case was to allow clinical researchers using commonly available observational datasets to conduct research efficiently using a common framework. In particular, the GDM was designed to allow researchers to construct their own visits and to reuse an extensive, published body of existing algorithms for identifying clinical research constructs which are expressed in the native vocabularies of the raw data. The GDM specifically considered both oncology research, which has its own specific vocabularies, and economic research. However, the model was designed so that these specific focus areas would not limit the design or use of the model.

3.1 Design goals

There were two primary goals in defining the standard tables and table relationships for the GDM, described below.

3.1.1 Use a single table for clinical codes in their original vocabularies

The first goal was make all clinical codes (e.g., International Classification of Diseases [ICD], Current Procedural Terminology, National Drug Codes, etc.) easy to load into the data model and to query because they represent the majority of electronic clinical information. This goal defined the key organizing structure of the GDM: the placement of all clinical codes in a central "fact" table. This is not unlike the i2b2 data model that uses a fact table to store all "observations" from a source data set; however, the GDM was not designed as a star schema despite the similar idea of locating the most important data at the center of the data model.

Furthermore, retaining the original (source) vocabulary prevents the loss of the original semantic expression of the underlying clinical information. Because of this decision for our logical data model, we created separate "details" tables to store information not shared among all records for drugs, inpatient hospitalizations, and measured variables. However, there is no requirement that the physical model conform to the logical model (see Discussion).

3.1.2 Retain hierarchical information with provenance

The second goal was to capture important hierarchical relationships among data elements within a relational data structure. Based on the review of numerous data sources, we decided on a two-level hierarchy for grouping clinical codes, with the lower level table called Contexts and the higher-level table called Collections. This was based on common data structures where many related codes are recorded on a single record in the source data (Contexts), and where

these records are often grouped together (Collections) based on clinical reporting or billing considerations. See Figure 1 for a visual depiction of the hierarchical structure.

Our review of data sources suggested that there were only a handful of relationships that were required to support this structure. The primary relationship represents data that is reported together or collected at the same time. One example of this includes a “line”, which occurs in claims data when one or more diagnosis codes, a procedure code, and a cost are all reported together. Another example includes laboratory values assessed at the same time (e.g., systolic and diastolic blood pressure) which could be considered to be co-reported. Also, a set of prescription refills could represent a linked set of records. Even records that contain pre-coordinated expressions (i.e., a linked set of codes used to provide clinical information akin to an English sentence) could also be stored in order as a single Context record.

We also included the provenance for each clinical code as part of Contexts, recording not only the type of relationship among elements within a Context as discussed above, but also the source file from which the data was abstracted. To minimize the loss of information when converting data to another data model, the GDM does not require visits to be created. This is important because visits are not consistently defined among other data models, particularly for administrative claims data. However, in electronic health record (EHR) datasets, visits may be better defined and therefore recorded in the Collections table. Visits can become confusing when considering costs, because costs may be present at a Context (line-item) or Collection (invoice) level. Therefore, this led us to align costs with the Contexts table. By evaluating the type of the context record (i.e., “claim” versus “line”) users can determine whether a cost is an aggregated construct or not. We adapted the Payer Reimbursements and the Costs tables from the OMOP version 5.20 cost table because it was the only data model with a cost table, and because we contributed substantially to its design. However, unlike the single OMOP cost table, we created two tables to accommodate both reimbursement-specific information, which has a well-defined structure, and all other kinds of economic information, which requires a very flexible structure.

3.1.3 Other Considerations

There are several other considerations made in building this data model, some of which were borrowed or adapted from other data models. For example, in addition to the cost table, we borrowed the OMOP idea to store all codes as “concept ids” (unique numeric identifiers for each code in each vocabulary to avoid conflicts between different vocabularies that use the same

code). We also adapted the idea of OMOP "type_concept_ids" to track provenance within our data model. Finally, we wanted to retain enrollment information so that the data could be used for different purposes (e.g., if a study does not require drug data, then enrollment in a drug plan should not be required). Although it was not an explicit requirement in designing the GDM, we hoped to facilitate a straightforward, subsequent ETL process to other data models, including OMOP, Sentinel, and PCORnet.

While this data model does not require any relationships (mappings) to standard vocabularies, it does include a Mappings table to make it easier to express relationships among data sources to the extent it is needed or desired. These mappings may be particularly useful in the context of automated analyses or standardized reports.

3.2 Test Data

We tested the data model on three very different types of commonly available data used by clinical researchers: administrative claims data, EHR data, and cancer registry data. Claims in the United States are generally submitted electronically by the provider to the insurer using the American National Standards Institute (ANSI) 837P and 837I file specifications, which correspond to the CMS-1500 and UB04 paper forms.[14] Remittance information is sent from the insurer to the provider using the 835P and 835I specifications. However, actual claims data used for research is provided in a much simpler format. Based on experience developing and supporting software for submitting claims to insurers as well as creating ETL specifications for multiple commercial claims and EHR datasets using the OMOP data model, we determined that Medicare data is the most stringent test for transforming claims data because it contains the most information from the 837 and 835 files. For EHR data, we used the Clinical Practice Research Datalink (CPRD) data, because it is widely used for clinical research.[15] Finally, as part of our focus on oncology research, we included Surveillance, Epidemiology, and End Results (SEER) data[16] because SEER provides some of the most detailed cancer registry data available globally to clinical researchers.

More specifically, we implemented a complete ETL process for the Medicare Synthetic Public Use Files (SynPUF). The SynPUF data are created from a 2.1-million-person sample of Medicare beneficiaries from 2008 who were followed for three years, created to facilitate software development using Medicare data.[17, 18] We also implemented an ETL for SEER data linked to Medicare claims data [19] for 20,000 patients with small cell lung cancer, as part of an ongoing research project to describe patterns of care in that population. Finally, we

developed a complete ETL for 140,000 CPRD patients for an ongoing research project evaluating outcomes associated with adherence to lipid-lowering medications. We also tested the feasibility of an ETL process to move SynPUF data from the GDM to the Sentinel data model (version 6.0) to ensure that the model did not contain any structural irregularities that would make it difficult to move data into other data model structures.

3.3 ETL Software

The ETL processes were built using R (version 3.2.5) and the data.table package (version 1.10.4).[20] R was selected because it is an open-source, cross-platform software package; because of its flexibility for composing ETL functions; and because of the availability of the data.table package as an in-memory database written in C for speed. The package itself is modular, and allows users to compose arbitrary ETL functions. Although the approach is different, the process is conceptually related to the dynamic ETL described by Ong, et al.[21]

4. Results

The resulting data model contains 19 tables (see hierarchical view in Figure 2). Details of the tables are provided in Supplementary Materials, and the most up-to-date version is available on a GitHub repository.[22]

4.1 Tables

The Clinical Codes, Contexts, and Collections tables make up the core of the GDM (as shown in Figure 1). All clinical codes are stored in the Clinical Codes table. Each row of the Clinical Codes table contains a single code from the source data. In addition, each row also contains a person id, the associated start and end dates for the record, a provenance concept id, and a sequence number. The sequence number allows codes to retain their order from the source data, as necessary. The most obvious example from billing data is diagnosis codes that are stored in numbered fields (e.g., diagnosis 1, diagnosis 2, etc.). But any set of ordered records could be stored this way. Grouping together ordered records is accomplished by associating them with the same id from the Contexts table. The provenance id allows for the specification of the type of record (e.g., admitting diagnosis, problem list diagnosis, etc.).

The Contexts table allows for grouping clinical codes and storing information about their origin. The record type concept id identifies the type of group that is stored. Examples might include lines from claims data where diagnoses, procedures, and other information are grouped, prescription and refill records that might be in electronic medical record or pharmacy data, or measurements of some kind from electronic health record or laboratory data (e.g., systolic and diastolic blood pressure, or a laboratory panel). In addition, the table stores the file name from the source data, the Centers for Medicare and Medicaid (CMS) place of service values (used for physician records since facility records to not have a place of service in claims data), and foreign keys to the care site and facility tables. The Contexts table also contains a person id and both start and stop dates which could be different from the start and stop dates of the individual records (e.g., a hospitalization may have different dates from the individual records within the hospitalization).

The Collections table represents a higher level of hierarchy for records in the Contexts table. That is, records in the Collections table represent groups of records from the Contexts table. This kind of grouping occurs when multiple billable units ("lines" or "details") are combined into invoices ("claims"). It also occurs when prescriptions, laboratory measures, diagnoses and/or procedures are all recorded at a single office visit. In short, a Collection is typically a "claim" or a

"visit" depending on whether the source data is administrative billing or electronic health record data. In using a hierarchical structure, the model avoids the need to construct "visits" from claims data which often leads to inaccuracy, loss of information, and complicated ETL processing. In the simplest possible case, it is possible to have a single record in the Clinical Codes table which is associated with a single Context record, which is associated with a single Collection record, as shown in Figure 1 for a drug record. The critical part of the ETL process – moving data into the Clinical Codes, Contexts, and Collections tables – is described in Figure 2 for the SynPUF data.

The Details tables capture domain-specific information related to hospitalizations, drugs, and measurements. The Admissions Details table stores admissions and emergency department information that doesn't fit in the Clinical Codes, Contexts, or Collections tables. It is designed to hold one admission per row. Each record in the Collections table for an inpatient admission links to this table.

There are two tables that store cost, charge, or payment data of some kind. The Payer Reimbursements table stores information from administrative claims data, with separate columns for each commonly used reimbursement element. All other financial information is stored in the Costs table, which is designed to support arbitrary cost types, and uses a "value_type_concept_id" to indicate the specific type. For both tables, each record is linked to a record in the Contexts table. In administrative claims data, this means that each "line" (diagnosis and procedure) can have a cost record. For records that have costs only at the claim/header level (e.g., inpatient hospitalizations), only Contexts that refer to "claims" (i.e., a record_type_concept_id for "claim") will have costs. For data with costs at both the line and claim/header level, costs can be distinguished by the Context type. In our experience, the sum of the line costs does not always equal the total cost, so depending on the research question, the researcher will need to determine whether claim, line, or both should be used. It is possible that each Clinical Code record sharing a single Contexts record could have a different cost; therefore, the two cost-related tables include a column to indicate the specific Clinical Code record to which the cost belongs. This might occur, for example, if multiple laboratory tests have different costs, but are share a common provenance (i.e., Contexts record).

The Facilities table contains unique records for each facility where a patient is seen. The facility_type_concept_id should be used to describe the whole facility (e.g., Academic Medical Center or Community Medical Center). Specific departments in the facility should be entered in

the Contexts table using the care_site_type_concept_id field. The Addresses table captures address information for patients, practitioners, and facilities.

The Contexts Practitioners table links one or more practitioners with a record in the Contexts table. Each record represents an encounter between a patient and a practitioner in a specific context. This role_type_concept_id in the table captures the role, if any, the practitioner played on the context (e.g., attending physician).

The Deaths table captures mortality information at the patient level, including date and cause(s) of death. This is commonly populated from beneficiary or similar administrative data associated with the medical record. However, it is useful to check discharge status in the Admissions Details table as part of ETL process to ensure completeness. There are also diagnosis codes that indicate death. Deaths that are indicated by diagnosis codes will be in the Clinical Codes table and should not be moved to the Deaths table. If needed, these codes can be identified using an appropriate algorithm (e.g., a set of ICD-9 codes) to identify death as part of the identification of outcomes in an analysis.

The Information Periods table captures periods of time during which the information in each table is relevant. This can include multiple records for each person, including records for different enrollment types (e.g., Medicare Part A, Medicare Part B, or Medicare Managed Care) or this can be something as simple as a single date range of “up-to-standard” data as provided by the Clinical Practice Research Datalink. This table includes one row per patient per enrollment type.

The Concepts table provides a unique numeric identifier (“concept_id”) for each source code in each vocabulary used in the data (see Table 1). The Concepts table does not have to be consistent across databases, although there may be efficiencies in using a consistent set of identifiers for all entries from commonly used vocabularies. The specific vocabularies used in the data are provided in the Vocabularies table. The idea of having both Concepts and Vocabularies tables was adapted from the OMOP data models. As mentioned in Methods, the Mappings table allows for the expression of consistent concepts across databases (consistency across databases is not explicitly required).

Table 1: Example Concept Table for Variables in the SEER Data

id	vocabulary_id	concept_code	concept_text
100010943	naaccr_grade	1	Grade I; grade i; grade 1; well differentiated; differentiated, NOS
100010944	naaccr_grade	2	Grade II; grade ii; grade 2; moderately differentiated; moderately well differentiated; intermediate differentiation
100010945	naaccr_grade	3	Grade III; grade iii; grade 3; poorly differentiated; differentiated
100010946	naaccr_grade	4	Grade IV; grade iv; grade 4; undifferentiated; anaplastic
100010947	naaccr_grade	5	T-cell; T-precursor
100010948	naaccr_grade	6	B-cell; Pre-B; B-Precursor
100010949	naaccr_grade	7	Null cell; Non T-non B;
100010950	naaccr_grade	8	N K cell (natural killer cell)
100010951	naaccr_grade	9	cell type not determined, not stated or not applicable
Note: SEER = Surveillance, Epidemiology and End Results, a collection of data from cancer registries across the United States. NAACCR = North American Association of Central Cancer Registries.			

The Mappings table is designed to express relationships among data elements. It can also be used to facilitate translation into other data models (see Table 2). In a few very simple cases like sex and race/ethnicity, the GDM encourages concept mappings to a core set of values to make it easier for users of a study-builder software applications to filter patients by age, gender, and race/ethnicity using a simpler representation of the underlying information. The Mappings table also permits an arbitrarily complex set of relationships, along the lines of the approach taken with the OMOP model and the use of standard concepts for all data elements. By using a Mappings table, we reduce the need to re-load the entire dataset when new mappings become available. Regardless of how the Mappings table is used, the GDM still retains the original codes from the raw dataset.

Table 2: Example Mapping Table for Conversion to Sentinel Common Data Model

concept_ id _1	relationship _id	concept _id _2	vocabulary _id_1	concept _code_1	concept _text_1	vocabulary _id_2	concept _code_2	concept _text_2
100000002	is_a	100000092	synpuf _sex	2	Female	sentinel _sex	F	Female
100000001	is_a	100000093	synpuf _sex	1	Male	sentinel _sex	M	Male
100000003	is_a	100000095	synpuf _race	1	White	sentinel _hispanic	N	Not Applicable
100000004	is_a	100000095	synpuf _race	2	Black	sentinel _hispanic	N	Not Applicable
100000005	is_a	100000095	synpuf _race	3	Others	sentinel _hispanic	N	Not Applicable
100000006	is_a	100000097	synpuf _race	5	Hispanic	sentinel _hispanic	Y	Yes
100000005	is_a	100000098	synpuf _race	3	Others	sentinel _race	0	Unknown
100000006	is_a	100000098	synpuf _race	5	Hispanic	sentinel _race	0	Unknown
100000004	is_a	100000101	synpuf _race	2	Black	sentinel _race	3	Black or African American
100000003	is_a	100000103	synpuf _race	1	White	sentinel _race	5	White

Note: This table shows how sex and race/ethnicity would be associated if the GDM were to use the Sentinel definitions as its defaults. Mappings allow users to query the data based on standardized default values for these variables while retaining the values from the original data. The GDM can support more extensive mapping relationships, but these are not required. Note that only the first 3 columns are part of the Mappings table. The remaining columns are created from joining to the Concepts and Vocabularies tables.

4.2 ETL Results

We loaded SynPUF data and SEER Medicare data into the GDM. After downloading the data to a local server, the process of migrating the SynPUF data with 2.1 million patients of data to the GDM took approximately 8 hours on a Windows server with 4 cores and 128 Gb of RAM and conventional hard drives (running two files at a time in parallel). Most of the time was spent loading files into RAM and writing files to disk since the process of ETL with the GDM is primarily about relocating data.

SEER Medicare data for SCLC included approximately 20,000 patients and took less than 1 hour. Selected SEER data was included in the ETL process ignoring recoded versions of

existing variables or variables used for consistency of interpretation over time. The ETL process focused on 31 key variables including histology, location, behavior, grade, stage, surgery, radiation, urban/rural status, and poverty indicators. Each SEER variable was included as a new vocabulary in the Concepts table (see Table 1).

CPRD data included approximately 140,000 patients and took approximately 2 hours. For the Test file which contains laboratory values and related measurements, we used Read codes in the Clinical Codes table; however, it would be trivial to add the “entity types” (numeric values for laboratory values and other clinical measurements and assessments) to the Clinical Codes table as well, with both the Read code and the entity type associated with the same Context record and the same Measurement Details record. We used the entity types for all records in the Additional Clinical Details table, although this could also be done differently. In all cases, the Mapping table allows for alternative relationships to be added to the data.

The process of moving data from the GDM into the Sentinel data model was straight-forward. Most of the process involved mapping Collections and Context records to visit types from Sentinel, and partitioning the information from the Clinical Codes table into the different domains specified in the data model (e.g., Procedure, Diagnosis, Dispensing, etc.).

5. Discussion

The GDM is designed to allow clinical researchers to identify the clinical and economic constructs needed for a wide range of research areas without side effects from the requirement to create visits or domains. This provides flexibility for researchers to study not only clinical encounters like visits, hospitalizations, emergency room visits, or even episodes of care, but also more basic constructs like conditions or medication use. Its main goal is to simplify the location of the most important information for creating analysis data sets and studies, which has the benefit of making ETL easier. It does this by using a hierarchical structure instead of the "visit-centric" approach taken by other data models. It tracks the provenance of the original data elements to enhance the reproducibility of studies. It includes a table to store relationships among data elements for standardized analyses. And it allows for a straightforward, subsequent ETL process to other common data models to make it easier for researchers to utilize the analytical tools and frameworks associated with those models.

Part of what makes ETL labor-intensive is that many data models store records in tables intended to hold specific types of information (i.e., domains) to ease analyses. Examples vary by the data model, but typical domains include conditions, procedures, drugs, laboratory values, and observations. Moving raw healthcare data into a data model requires either forcing each vocabulary (e.g., Read) into a single domain, or mapping each code within each vocabulary to its specific domain. Placing codes in domain-specific tables can be particularly challenging when vocabularies cross domains (e.g., Read and ICD). The GDM does not use domains; instead, all clinical codes are placed in a single table. If needed, the GDM could support domain information in the Mappings table through the use of a "has_domain" relationship. It could be argued that it is easier to change a domain mapping by updating the Mappings table, rather than re-building the entire data set to move records among the various tables.

Because other data models use visits to connect information within the data model, our emphasis on avoiding visits deserves comment. Visits are seldom required for clinical research, unless the enumeration of explicit visits is the research topic itself, as might be the case in some health services research. However, for a large portion of research projects, the focus is on identifying the dates of clinically relevant interactions with the healthcare system, which does not require knowledge of a visit, per se. It is a research project in and of itself to define visits, and such definitions are specific to the research question being investigated.[13] A data model should allow researchers to define visits as appropriate to answer their research questions.

The challenges with visits can best be seen by inspecting the guidelines for creating visits from each data model. In the Sentinel version 6 data model[10], a visit is defined as a unique combination of patient, start date, provider and visit type. Visit types are defined as Ambulatory, Emergency Department, Inpatient Hospital, Non-acute Institutional, and Other. Furthermore, "Multiple visits to the same provider on the same day should be considered one visit and should include all diagnoses and procedures that were recorded during those visits. Visits to different providers on the same day, such as a physician appointment that leads to a hospitalization, should be considered multiple encounters."

PCORnet version 3.0 is very similar to Sentinel.[12] However, PCORnet provides additional guidance on emergency department records, and provides a visit type for emergency department visit that becomes an inpatient stay. It also allows for visit types of Other Ambulatory, No Information, and Unknown.

In the OMOP version 5.20 data model, a visit is defined for each "visit to a healthcare facility." According to the specifications[6], in any single day, there can be more than one visit. One visit may involve multiple providers, in which case the ETL must either specify how a single provider is selected or leave it null. One visit may involve multiple care sites, in which case the ETL must either specify how a single site is selected or leave it null. Visits must be given a type, which is limited to Inpatient Visit, Outpatient Visit, Emergency Room Visit and Long Term Care Visit.

For i2b2, the specifications state a visit ". . . can involve a patient directly, such as a visit to a doctor's office, or it can involve the patient indirectly, as in when several tests are run on a tube of the patient's blood. More than one observation can be made during a visit. All visits must have a start date / time associated with them, but they may or may not have an end date. The visit record also contains specifics about the location of the session, such as the hospital or clinic the session occurred and whether the patient was an inpatient or an outpatient at the time of the visit." There are no specified visit types, and the data model allows for an "unlimited number of optional columns but their data types and coding systems are specific to the local implementation."[4]

Clearly, each data model has different perspectives on the definition of a visit. Such ambiguity can lead to differences in how tables are created in the ETL process. As a result, inconsistencies within or across data models can lead to differences in results, an issue that has already been demonstrated.[23] Challenges related to visits include using visits to connect records, like procedures with their associated conditions; separating emergency department

records from their associated inpatient admissions; and classifying physician records from inpatient hospitalizations to either inpatient or outpatient/ambulatory visits (most data models consider physician records to reflect outpatient or ambulatory care, regardless of the place of service). Laboratory records could be visits as with i2b2, or could be associated with visits as with other data models. Similarly, prescriptions and refills could be considered visits, or associated with visits. And other information, like family history, might not require a visit at all. In short, the most important structural component of other data models cannot be accurately and consistently defined, which creates problems with consistent analyses across the data models.

We should also note that the GDM describes the logical model, and not necessarily the physical model for the data. Information from other tables (e.g., the “Details” tables, and/or the Contexts and Collections tables, see Figure 2) could easily be combined into a larger Clinical Codes table for a more de-normalized structure without any loss of fidelity. The benefits of de-normalization depend primarily on the database technology being used. On a related note, the GDM logical model may be simpler for humans to use as well, because the most important clinical information is in a single table. The ability for humans to understand and access data using the data model is important for conceptualizing and debugging analyses using the data.

The GDM can be used as a standardized waypoint in a data pipeline because all necessary information should be available for the subsequent migration of data into other data models. Other data models have substantial user bases and useful analytical tools for conducting research. By starting with the GDM, researchers can implement modest transformations to migrate their data to other data models and make use of the analytical tools offered by those models. While this requires another ETL process, or a view to be created on the GDM, it facilitates access to useful analytical tools.

There are limitations to the model. While we have tested it against data that is commonly used by health services researchers and epidemiologists, there are likely to be specific data sets that will require modifications or improvements. The GDM does not yet include tables for genomic data or free text notes which are becoming more widely available for researchers. Also, more detailed location information may need to be added for those with access to additional data (which is often limited due to privacy issues).

5.1 Conclusion

The GDM is designed to retain the relationships among data elements to the extent possible, facilitating ETL and study building as part of a complete data pipeline for clinical researchers using commonly available observational data. Furthermore, it offers researchers a simpler process of standardizing the location of data in a defined structure and may make it easier for users to transform their data into other data models.

6. Declarations

6.1 Ethics approval

Not applicable because no clinical data was analyzed or reported.

6.2 Consent for publication

Not applicable.

6.3 Availability of data and materials

The data model is publicly available. The raw data is not available due to privacy reasons, except for the Synthetic Public Use data. See references for specific hyperlinks.

6.4 Funding

This research was self-funded.

6.5 Acknowledgements

We gratefully acknowledge the influence of the open-source OMOP common data model specifications on our thinking in creating our data model. In addition, we acknowledge the influence of Sentinel, PCORnet, and i2b2 on our approach, although most of our data model was designed prior to reviewing these models in detail. We also thank Chris Adamson for helpful discussions about organizing the data model in different ways. At the time of writing, all references to the concepts table refer to the OMOP version 5.20 vocabulary table maintained by OHDSI. However, there is no reason that a user could not create their own system of codes with unique identifiers across vocabularies, or use the codes from the National Library of Medicine Metathesaurus.[24]

6.6 Authors' information

MD is an epidemiologist who has worked with a wide variety of clinical data sources across therapeutic areas. JD and RD have extensive experience designing software for providers to submit medical bills to insurers. JD has constructed and/or substantially revised OMOP ETL specifications for many commercially available data sources. MD, JD, and RD are collaborators in the Observational Data Health Sciences and Informatics organization.

6.7 Authors' contributions

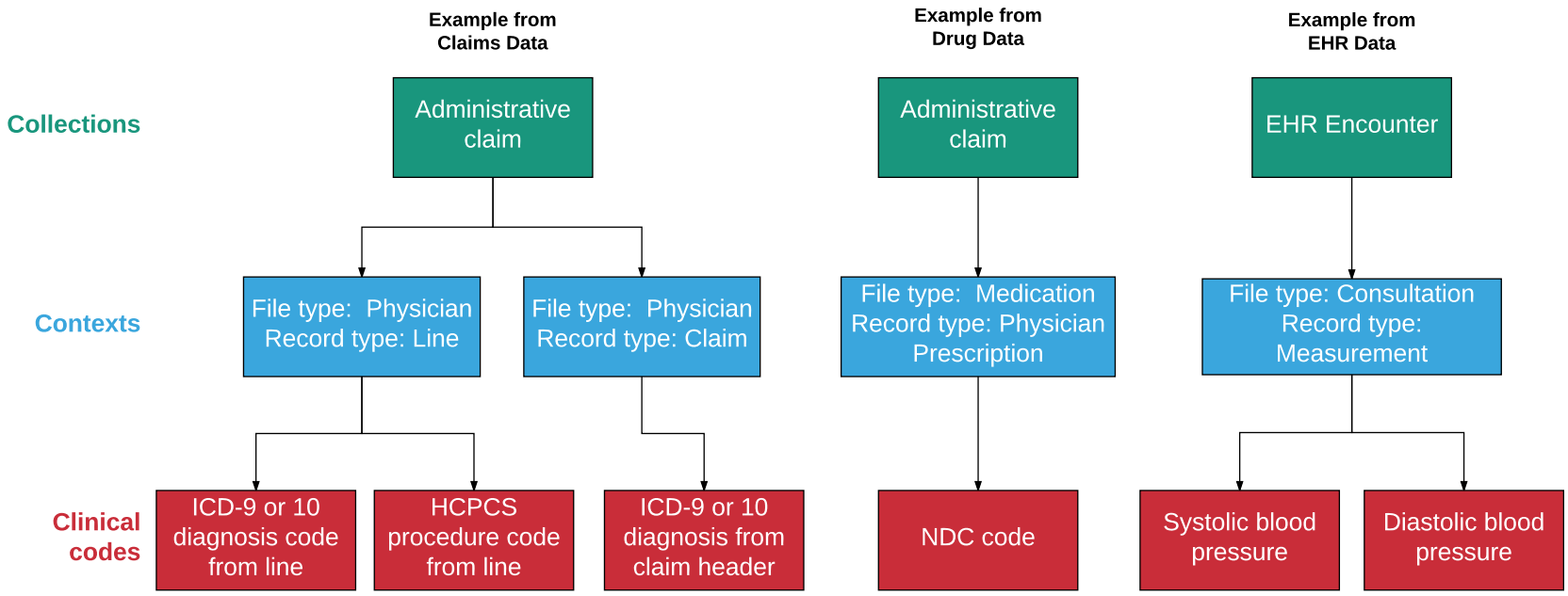
MD, MH, RD, and JD contributed to the design of the data model. MH developed all ETL code and ran all tests.

6.8 Competing interests:

Outcomes Insights provides consulting services and software for building studies using observational data.

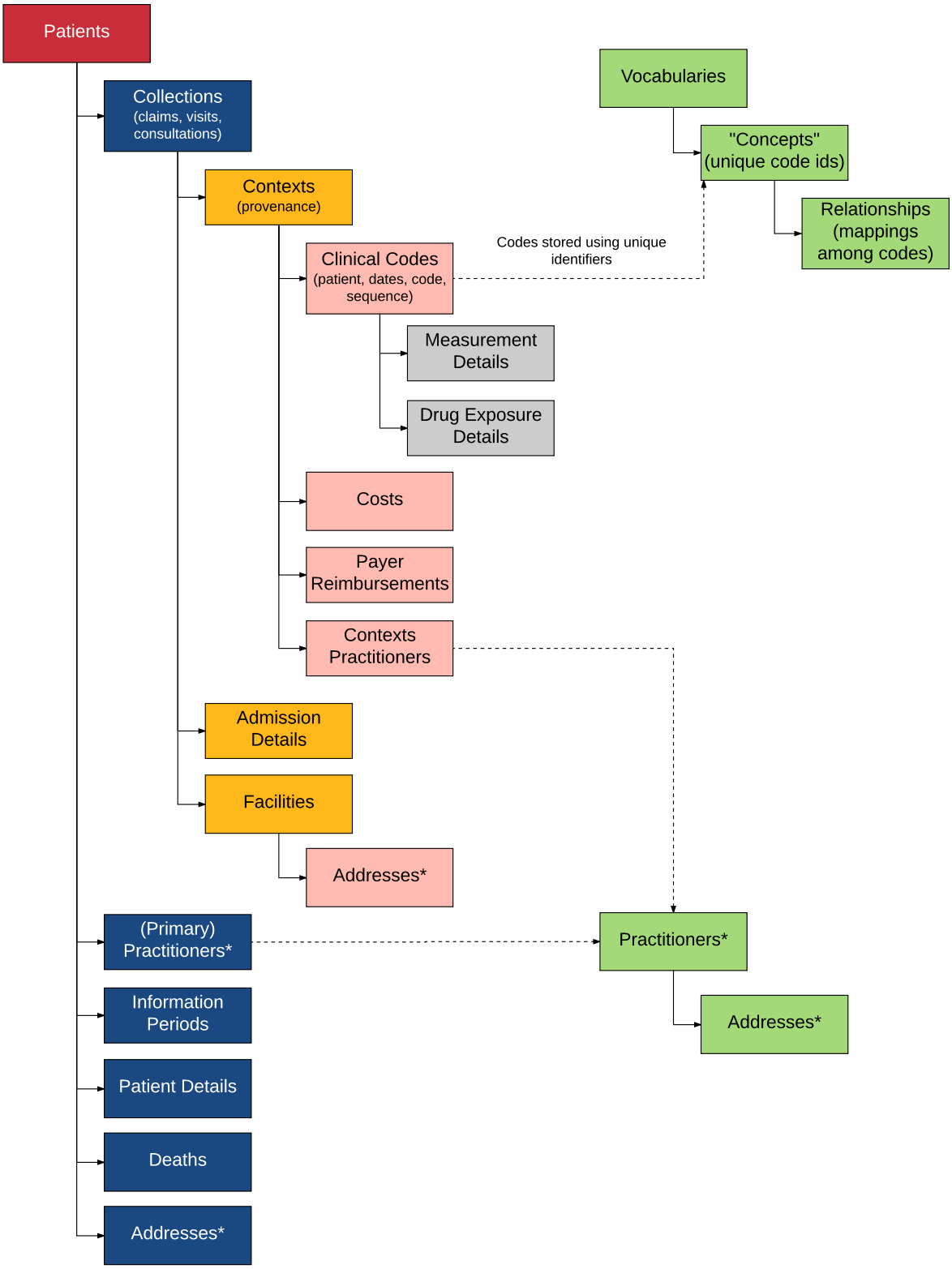
7. Figures

Figure 1: Relationships Among the Collections, Contexts, and Clinical Codes Tables



Note: EHR = electronic health record. HCPCS = Healthcare Common Procedure Coding System. NDC = National Drug Code. ICD = International Classification of Diseases. Figure does not contain specific data, but is intended to show the conceptual relationships among data elements across tables.

Figure 2: Hierarchical View of the Generalized Data Model



Note: Table names and key contents are listed above showing the key relationships among tables. See Supplementary Materials for more detail on tables. Tables in green serve as lookup tables across the database, including the vocabularies used. There is a single addresses table for unique addresses, and a single Practitioners table for unique practitioners. The Contexts Practitioners table allows multiple practitioners to be associated with a Context record.

Figure 3: Visualization of the ETL Process for SynPUF Data

SynPUF Record

patient_id	clm_from_dt	clm_thru_dt	icd9_dgns _cd_1	icd9_dgns _cd_2	hcpcs _cd_1	hcpcs _cd_2	line_icd9 _dgns_cd_1	line_icd9 _dgns_cd_2
54916	2009-10-14	2009-10-14	716.84	719.44	73218	73564	719.44	719.44

Clinical Codes

id	collection_id	context_id	patient_id	start_date	end_date	clinical_ concept_id	quantity	seq_num	type_ concept_id	clinical_code_ source_value	clinical_code_ vocabulary_id
8632447	10	3132545	54916	2009-10-14	2009-10-14	44831496		1	100000075	716.84	ICD9CM
8987856	10	3132545	54916	2009-10-14	2009-10-14	44829216		2	100000075	719.44	ICD9CM
18754452	10	4741345	54916	2009-10-14	2009-10-14	44829216		1	100000075	719.44	ICD9CM
6642545	10	4741345	54916	2009-10-14	2009-10-14	2211465		1	100000075	73218	CPT4
23495395	10	9482288	54916	2009-10-14	2009-10-14	44829216		2	100000075	719.44	ICD9CM
4740953	10	9482288	54916	2009-10-14	2009-10-14	2211482		2	100000075	73564	CPT4

Contexts

id	collection_id	patient_id	start_date	end_date	facility_id	facility_ type_ concept_id	pos_ concept_id	file_ type_ concept_id	address_id	service_ specialty_ type_	type_ concept_id
3132545	10	54916	2009-10-14	2009-10-14				100000062			100000064
4741345	10	54916	2009-10-14	2009-10-14				100000062			100000065
9482288	10	54916	2009-10-14	2009-10-14				100000062			100000065

Collections

id	patient_id	start_date	end_date	facility_id	admission_detail_id
10	54916	2009-10-14	2009-10-14		

Note: Clinical codes are derived from a single row in the source data set (SynPUF record). Colored arrows indicate how each group of codes is used to create records. Each code from the original record gets its own row in the Clinical Codes table. Codes that are

grouped together (e.g., line diagnosis 1 and procedure 1 in yellow) share the same context. In the Contexts table, type concept id ending in “64” indicates a claim level context, and the id ending in “65” indicates a line level context. The three contexts (groups of codes) share the same collection id.

8. References

1. Kahn MG, Batson D, Schilling LM. Data Model Considerations for Clinical Effectiveness Researchers. Med Care. 2012;50:S60–7.

2. Klann JG, Abend A, Raghavan VA, Mandl KD, Murphy SN. Data interchange using i2b2. J Am Med Informatics Assoc. 2016;23:909–15.

3. Murphy SN, Weber G, Mendis M, Gainer V, Chueh HC, Churchill S, et al. Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). J Am Med Informatics Assoc. 2010;17:124–30.

4. i2b2 Common Data Model. https://i2b2.org/software/files/PDF/current/CRC_Design.pdf. Accessed 20 Apr 2017.

5. Overhage JM, Ryan PB, Reich CG, Hartzema AG, Stang PE. Validation of a common data model for active safety surveillance research. J Am Med Inform Assoc. 2012;19:54–60.

6. OHDSI. OMOP Common Data Model. http://www.ohdsi.org/web/wiki/doku.php?id=documentation:overview. Accessed 20 Apr 2017.

7. Voss EA, Makadia R, Matcho A, Ma Q, Knoll C, Schuemie M, et al. Feasibility and utility of applications of the common data model to multiple, disparate observational health databases. J Am Med Informatics Assoc. 2015;22:553–64.

8. Psaty BM, Breckenridge AM. Mini-Sentinel and regulatory science--big data rendered fit and functional. N Engl J Med. 2014;370:2165.

9. Curtis LH, Weiner MG, Boudreau DM, Cooper WO, Daniel GW, Nair VP, et al. Design considerations, architecture, and use of the Mini-Sentinel distributed data system. Pharmacoepidemiol Drug Saf. 2012;21 SUPPL. 1:23–31.

10. Sentinel Common Data Model. https://www.sentinelinitiative.org/sentinel/data/distributed-database-common-data-model. Accessed 20 Apr 2017.

11. Fleurence RL, Curtis LH, Califf RM, Platt R, Selby J V, Brown JS. Launching PCORnet, a national patient-centered clinical research network. J Am Med Inform Assoc. 2014;21:578–82.

12. PCORnet Common Data Model. http://www.pcornet.org/pcornet-common-data-model/. Accessed 20 Apr 2017.

13. Tyree PT, Lind BK, Lafferty WE. Challenges of using medical insurance claims data for utilization analysis. Am J Med Qual. 21:269–75.

14. Centers for Medicare and Medicaid Services. Medicare Fee-for-Service Companion Guides. https://www.cms.gov/Medicare/Billing/ElectronicBillingEDITrans/CompanionGuides.html. Accessed 24 Oct 2017.

15. Herrett E, Gallagher AM, Bhaskaran K, Forbes H, Mathur R, van Staa T, et al. Data Resource Profile: Clinical Practice Research Datalink (CPRD). Int J Epidemiol. 2015;44:827–36.

16. Park HS, Lloyd S, Decker RH, Wilson LD, Yu JB. Overview of the Surveillance, Epidemiology, and End Results database: evolution, data variables, and quality assurance. Curr

Probl Cancer. 36:183–90.

17. Danese MD, Voss EA, Duryea J, Gleeson M, Duryea R, Matcho A, et al. Feasibility of Converting the Medicare Synthetic Public Use Data Into a Standardized Data Model for Clinical Research Informatics. In: AMIA 2015 Annual Symposium. San Francisco; 2015.

18. Centers for Medicare and Medicaid Services. Synthetic Public Use File. https://www.cms.gov/research-statistics-data-and-systems/downloadable-public-use-files/synpufs/. Accessed 20 Apr 2017.

19. Warren JL, Klabunde CN, Schrag D, Bach PB, Riley GF. Overview of the SEER-Medicare data: content, research applications, and generalizability to the United States elderly population. Med Care. 2002;40 8 Suppl:IV-3-18.

20. Comprehensive R Archive Network. R.

21. Ong TC, Kahn MG, Kwan BM, Yamashita T, Brandt E, Hosokawa P, et al. Dynamic-ETL: a hybrid approach for health data extraction, transformation and loading. BMC Med Inform Decis Mak. 2017;17:134.

22. Outcomes Insights Inc. Generalized Data Model. https://github.com/outcomesinsights/generalized_data_model. Accessed 20 Apr 2017.

23. Xu Y, Zhou X, Suehs BT, Hartzema AG, Kahn MG, Moride Y, et al. A Comparative Assessment of Observational Medical Outcomes Partnership and Mini-Sentinel Common Data Models and Analytics: Implications for Active Drug Safety Surveillance. Drug Saf. 2015;38:749–65.

24. Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical terminology. Nucleic Acids Res. 2004;32 suppl 1:D267–70.