MIMIC in the OMOP Common Data Model*

Nicolas PARIS, *WIND-DSI, AP-HP, Paris, France* **AND Adrien PARROT,** *WIND-DSI, AP-HP, Paris, France*
(nicolas.paris@aphp.fr) (adrien.parrot@caramail.fr)

AP-HP, Paris, France

Objectives : In the age of big data, the intensive care unit (ICU) is very likely to benefit from real-time computer analysis and modeling based on close patient monitoring and Electronic Health Record monitoring. MIMIC is still the first open access database in the ICU. Many studies have shown that common data models (CDMs) improve database searching by allowing code, tools and experience to be shared. OMOP CDM is spreading all over the world. We have transformed MIMIC into OMOP (MIMIC-OMOP) and assess the quality of the transformation, the benefits for analysts and the gains and potential for community contributions.
Material & Method: A documented, tested, versioned, exemplified and open repository has been put in place to support the transformation and enhancement of community source code. The resulting data set was evaluated over a 48-hour datathon with 160 participants.
Result: Most of the data correspond to the model and much of the terminology has been standardized with an investment of 2 people for 500 hours. The model demonstrated its ability to support community contributions and was well received during the datathon with 15,000 requests executed with a maximum duration of one minute. *Conclusion*: The resulting MIMIC-OMOP data set is ready for replicable research and as this is the first freely available OMOP data set with actual data depersonalized expectations are for increased and improved collaborations for ICUs, generalizable to other services.

0 INTRODUCTION

Intensive care units (ICUs) are care units where the demand for care increases[**?**] while mortality reaches up to 30%, which is a major health problem[**?**]. Studies have shown that intensivists use a limited level of evidence to guide decision making[**?**] and that medical practices are sparse and variable. Knowing that the ICU patient health record is very detailed and that there is a high density environment for data production is a paradox. The increasing adoption of electronic health record (EHR) systems around the world is capturing large amounts of clinical data[**?**] and data mining has the potential to play an important role in clinical medicine[**?**]. Indeed, based on important medical informations, expectations are to improve clinical outcomes and practices, enable personalized medicine and guide early warning systems, and also easily enroll a large, multi-center cohort while minimizing costs.

MIMIC (Medical Information Mart for Intensive Care) is a 10 year semi-automatic dataset of over 60,000 intensive care stays with very high granularity (including EKG) from two successive intensive care information systems (CCI) at the Beth Israel Deaconess Medical Center in Boston. It is the first ICU database available free of charge and has been the subject of more than 300 international publications. However, its monocentric nature makes it difficult to generalize findings to other ICUs. The MIMIC relational data model reflects the original CCI, as evidenced by the two separate *inputevent_mv* and *ouputevent_cv* [**?**] or the two separate terminologies for physiological data. This leads analysts (datascientists, statisticians, etc.) to reconcile this heterogeneity when pre-processing each study.

Some studies have shown that using a common data model (CDM) by generalizing the structural (data model) and conceptual (terminological model) design database allows for multicentre research, exploitation of rare diseases and catalyzes research by sharing practices, source code and tools [**?**, **?**]. As Kahn and Al said [**?**]kahn-data-2012), "databases modelling is the process of determining how data are to be stored in a database". It specifies data types, constraints, relationship and metadata definitions and provides a standardized way to represent resources/data and their relationships. However, some studies have shown that the results are not fully reproducible from one CDM to another [**?**] or from one centre to another [**?**]. The lighter approaches argue that maintaining the local conceptual model [**?**] or the original conceptual and structural model [**?**] of the research database leads to better results. On the one hand, keeping MIMIC on its specific form will not solve the limitation for multicentric research and, on the other

*To whom correspondence should be addressed Tel: +33111111111; e-mail: nicolas.paris@aphp.fr

hand, a fully standardized form would introduce other disadvantages. The ideal solution is probably in between to allow local or standardized analysis depending on the research question.

OMOP (Observational Medical Outcomes Partnership Common Data Model) is a CDM originally designed for multi-centre drug-related adverse events and now extends to medical, clinical and genomic cases. OMOP provides structural and conceptual models such as SNOMED for diagnostics, RxNORM for drugs and LOINC for laboratory results. Several examples of database transformation to OMOP have been published [**?**, **?**] and OMOP stores 682 million patient records from around the world [**?**]. Each clinical area is stored in different dedicated tables. The OMOP conceptual model is based on a closure table pattern [**?**] capable of ingesting any simple, hierarchical and also graph terminologies such as SNOMED-CT. In addition to local terminologies, OMOP specifies and maintains a set of standard terminologies to be mapped unidirectionally (local to standard) by implementers. Although OMOP has proven its reliability [**?**], the concept mapping process is known to have an impact on results [**?**] and the application of the same protocol on different data sources leads to different results [**?**]. This shows the importance of keeping local codes so that local analysis is always possible. Previous preliminary work has been done on the translation of MIMIC into OMOP [**?**]. This work remains to be refined and updated to be evaluated.

In a recent CDM comparative study [**?**, **?**] OMOP obtained better results in the criteria of the evaluation database compared to the other models: completeness, integrity, flexibility, simplicity of integration and implementability, adapting to the wider coverage of standard terminologies, providing a more systematic analysis with an analytical library and visualization tools, providing SQL models easier to use. That is why OMOP offers a broader set of standardised concepts. In terms of structural CDM, OMOP is very rigorous in how data should be loaded into a particular table when i2b2 for example is very flexible with a general table that solves all data domains. This rigorous approach is necessary for standardization. Previous work has loaded i2b2 with MIMIC-III [**?**] - however, the concept mapping step has limited the results since i2b2 design does not store local ontologies or informations where OMOP design keeps concept mapping unfinished. OMOP has the advantage of not making the terminology mapping step mandatory by keeping local codes in a usable format. Compared to the FHIR, OMOP performs better as a conceptual CDM because the FHIR does not specify the terminology to be used. In terms of structural CDM, the OMOP relational model can be materialized in csv format and stored in any relational database when FHIR uses json files and needs some processing and more skills to exploit. We believe OMOP shares the advantages of all the above models.

In order to evaluate the transformation of MIMIC into OMOP, we propose to answer the following questions, such as the difficulty of transforming/maintaining an OMOP dataset from an local database, how the initial data is integrated and how much data is lost in the process, how the model should be queried simply and efficiently by analysts, how the design should be enriched by collaborative work, and finally to what extent OMOP can integrate and feed back to intensivists in a real-time context. This work is then evaluated according to 3 axes: Transformations, Contributions and Analyses. The *first* major contribution of this study is to evaluate OMOP in a freely accessible and well known database. The *second* major contribution is to provide a freely accessible dataset in OMOP format that could be useful to researchers. The *third* major contribution is to provide the OMOP community with useful transformations dedicated to intensive care that can be reused on any OMOP data set.

1 MATERIAL & METHOD

1.1 Data Transformation

All **transformation processes** are freely accessible to the public via the github website [**?**] maintained by MIT-LCP [**?**]. The repository is based on git and is designed for sharing, improvement, collaboration and reproducible work. Indeed, github is archived on a universal and durable software archive solution [**?**]. The github repository centralizes the various resources of this work such as documentation, source code, unit tests, as well as questioning examples, discussions and problem issues. It also indicates web resources such as the physical data model for MIMIC[**?**] and OMOP[**?**] datasets and the Achilles' web client[**?**].

The vast majority of source code is implemented in PostgreSQL 9.6.9 (PgSQL) because it is the primary support for the MIMIC database and allows the community to run our work on limited resources without needing a license. Finally, PgSQL has recently made enormous efforts to better manage data processing. Some elaborate data transformations have been implemented as PgSQL functions.

MIMIC-III version 1.4.21 (MIMIC) has been loaded with the scripts provided in a PgSQL instance. The OMOP CDM version 5.3.3.1 (OMOP) target tables were created from the provided scripts with some small changes stored in the change script. OMOP which defines 15 standardized *clinical* data tables, 3 *health* system data tables, 2 *health economics* data tables, 5 tables for *derived* elements and 12 tables for standardized *vocabulary*. We didn't use the health economics data tables (not provided by MIMIC). Indexes that would have slowed data migration with unnecessary calculations have been removed. Integrity constraints (primary keys, foreign keys, non-nullable columns) have been included to apply integrity checks at runtime. A subset of 100 patients was selected based on their broad representativeness of the database and cloned into a second instance to serve as a light and representative development set. Each source table has been added a global unique sequence incremented from 0 that serves as the primary key and link in the OMOP target tables.

Extract-Transformation-Load (ETL) is a methodology for migrating data from a source to a target location.

ETL first extracts the data from the source location, then applies the transformations to a dedicated computer and finally loads the resulting data into the target location. Extract-Load-Transform (ELT) processes are slightly different methodologies that does not use a dedicated server transformation. The data is extracted and loaded directly into the target location and subsequently transformed into the location.

The ELT is composed of PgSQL scripts, each extracting information from the source or concept mapping tables, then transforming and loading an OMOP target table. The order of these scripts is important and is done sequentially through a main script.

Each ELT part has been tested using pgTAP, a unit test framework for PgSQL. This allows you to check for loss of information, or code regression during development. Each unit test script checks whether a particular OMOP target table is loaded correctly - most tables are covered and tests cover simple counts, aggregate counts or distribution checks.

All character type columns with limited length have been modified as follows unlimited since it could cause unpredictable truncation of content, and it has no negative impact on PgSQL storage size or performance. The *visit_occurrence* and the *visit_detail* tables have been corrected accordingly some discussions on the OHDSI forum. The *nlp_note* table has been completed by fields corresponding to the online documentation. The character *offset* column has been divided into two integer type columns because the offset word is a SQL reserved word and it makes sense to fill the resulting *offset_begin* and *offset_end* resulting columns.

The **structural transformation** took place in several phases. The first phase consists of looping each MIMIC table and choosing an equivalent location in OMOP for each column. In general, the MIMIC documentation and the OMOP documentation were sufficiently informative. In several cases, we needed clarification from MIMIC contributors on the dedicated github repository, or from the OMOP community on the dedicated forum. All choices have been discussed in the repository [**?**] and can be tracked in the commit log.

For the second step, we tried not to infer any results. For laboratory tests when it makes sense to put a specimen (i.e. a body sample) for many laboratory results (because one blood sample can be used for several tests), we decided to create as many rows of samples as laboratory tests because the information is not present in MIMIC. The same was true when date information was not provided (*start /end_datetime* for *drug_exposure*). Chartevents and labevents tables provide many number fields as a string, which is not practical for statistical analysis. We provide a standard and easy enhancement by the community model to extract the numerical value of the string with a PgSQL function. The results of the MIMIC laboratory have been restructured to adapt to OMOP format. In particular, the numerical value (value_as_number) is accompanied by a mathematical operator (concept_operator_id) and a unit of measurement (concept_unit_id). All lines marked in error have not been converted to OMOP format since the MIMIC team plans to delete them at the next release.

By design, MIMIC aggregates information from various systems. Thus, the transfer information is divided into several tables, such as *admissions*, *transfers* and *icustays*. OMOP centralizes this information in the detail of the *visit_detail*. We added emergency stays as a normal location for patients throughout their hospital stay (unlike what had been done by MIMIC). Icustays raw mimic table has been removed because it is a table derived from the transfer table [**?**] and we decided to assign a new *visit_detail* for each ICU stay (based on the transfer table) while mimic preferred to assign a new icustay stay if a new admission occurs ¿ 24h after the end of the previous stay.

The **conceptual transformation** uses OMOP vocabulary tables that have been loaded from an Athena export [**?**] of all terminologies without license limitations.

Local MIMIC codes are also loaded into the concept table with a *concept_id* identifier from 2.1 billion (below this number is reserved for OMOP terminologies [**?**]). MIMIC codes can be distinguished with *the vocabulary_id* identifier equal to "MIMIC code" and a *domain_id* identifier targeting the OMOP table in which the corresponding data is stored. Later, this domain information is used in the ELT to send the information in the proper table. As the OMOP model did we adopt a "concept-driven methodology", domain of each local concept drive the concept to the right table.

Where possible, relevant information from the original MIMIC tables has been concatenated in the *concept_name* column. New local MIMIC concepts were introduced and given a value from 2 billion to distinguish them from local MIMIC concepts.

When it came to standardizing local MIMIC codes in OMOP standards codes, there were four distinct cases. In the *first* case, MIMIC is by chance already in OMOP standard terminology (e.g. LOINC laboratory results) and, therefore, the standard and local concepts are the same. In the *second* case, MIMIC is not in the standard OMOP terminology, but the mapping is already provided by OMOP (ex: ICD9/SNOMED-CT), so the domain tables have been loaded accordingly. In the *third* case, mapping is not provided, but it is small enough to be done manually in a few hours (such as demographic status, signs and symptoms). In the *fourth* case, mapping is not provided and terminology is enormous (admission diagnosis, drugs). Then, only a subset of the most represented code was manually mapped.

When the mapping concept is required manually, a mapping csv file has been built. This solution can be adapted to medical users who do not have training in database engineering. The spreadsheet has several columns such as local/standard labels, ids and also comments, evaluation metrics and a script loads them into the PgSQL when completed. In order to catalyse the mapping process, the language algorithm has proven to be effective [**?**] although OHDSI provides USAGI [**?**]. We have chosen to use simple SQL queries that are flexible enough to be queried on de-

mand or to generate a pre-filled csv with the best matches. It uses PGSQL full-text ranking features and links local and standard candidates with a rating function based on their labels. This work was followed by a intensivist check.

Although various types of information are stored in the measurement table, the dedicated OMOP concepts for the *measurement_type_concept_id* column were not sufficient to distinguish them. We have added some.

The actual *visit_detail* table does not introduce relevant information and duplicate informations from *visit_occurrence* table. For *admitting_concept_id* and *discharge_to_concept_id* columns, we extended the dictionary to track bed transfers and room transfers. For *visit_type_concept_id* column we assigned a new concept for any level of granularity necessary for your use case (ward, bed. . .)

1.2 Contribution

MIMIC provides a large number of SQL scripts to calculate derived scores and define cohorts. Some of them have been implemented in OMOP format and fill OMOP cohort tables. Common derived information was introduced and loaded: corrected serum calcium, corrected serum potassium, P/F ratio, corrected osmolarity, SAPSII.

A set of *general denormalized* tables has been built on top of the original OMOP format that have the *concept_name* related to the *concept_id* columns. The concept table is a central element of OMOP and, therefore, it is involved in many joins to obtain the concept label. Normalized tables accelerate calculation time and provide an easier set of data for analysis.

In addition, a set of *specialized materialized analysis views* has been built on the original OMOP format. Microbiologicalevents table is a reorganization of the measurement table datas of microorganisms and associated susceptibility testing antibiotics and is based on the MIMIC *microbiologicalevents* table. The OMOP icustays table allows to quickly obtain the patients admitted in resuscitation and is inspired by the MIMIC *icustays* tables.

The *note_nlp* table was originally designed to store final or intermediate derived information and metadata from clinical notes. When definitive, the extracted information is intended to be moved to the dedicated domain or table and then reused as regular structured data. When the information is still intermediate, it is stored in the *note_nlp* table and can be used for later analysis. To assess this table, we provided two information extraction pipelines. The *first* pipeline extracted numerical values such as weight, height, body mass index and left ventricular cardiac ejection fraction from medical notes with a SQL script. The resulting structured numerical values were loaded into the measurement or observation tables according to its domain. The *second* pipeline *section extractor* based on the apache UIMA framework divides notes into sections to help analysts choose or avoid certain sections of their analysis. While some methods already exist to extract medical sections [**?**], the prior work of describing sections was too high, and we opted for a naive approach. Section templates (such as "Illness History") have been automatically extracted from text with regular expressions, then filtered to keep only the most frequent (frequency ¿ to 1%). 1200 sections were collected and then manually filtered to exclude false positives. 400 similar groups were highlighted. The extracted sections have not been mapped to standard terminology such as LOINC CDO. The reason for this is that the CDO LOINC decided to delete its sections from its standard, considering that these sections were not widely used [**?**].

1.3 Data Analytics

A 48-hour open access datathon [**?**] was set up in Paris AP-HP (Assistance Publique des Hopitaux de Paris) once the MIMIC-OMOP transformation was ready for research to evaluate OMOP as an alternative data model in a real event. This datathon was organised in collaboration with the MIT. Scientific questions had been prepared in an online forum. Participants could introduce themselves and propose a topic or choose an existing one. OMOP has been loaded into apache HIVE 1.2.1 in ORC format. Users had access to the ORC dataset from a web interface jupyter notebooks with python, R or scala. A SQL web client allowed teams to write SQL from presto to the same dataset. The hadoop cluster was based on 5 computers with 16 cores and 220GB of RAM memory. The MIMIC-OMOP dataset has been loaded from a PGSQL instance to HIVE thought apache SQOOP 1.4.6 directly in ORC format. Participants also had access to the Schemaspy database physical model to access the OMOP physical data model with both table/column comments and key primary/foreign relationships materializing the relationships between the tables. All queries were been logged.

2 RESULT

2.1 Data Transformation

The MIMIC to OMOP conversion was performed by two developers (a data engineer and an intensivist) for an estimated 500 hours. This includes ELT, git documentation, concept mapping, contributions and unit tests. ELT (with unit tests and generation of ready-to-load archives) on the subset of 100 patients takes five minutes and enables rapid development cycles. On all MIMIC data, the ELT lasts 3 hours. The resulting csv archive is about the same size as the original archive, and it is also the same once instantiated in PGSQL and indexed.

The OMOP-CDM contains 37 data tables. We populated 19 tables. From MIMIC, we create a standardized model called MIMIC-OMOP.

The evaluation of a system and a structural model is rather difficult [**?**] but we have tried to evaluate it through several axes.

The first axe was the unit tests. During the all ETL process we created a lot of unit tests thanks to pgTap library. All are available on our github [**?**]. All the test passed.

The second axe was Achilles evaluation. Like many previous authors, we used the Achille software to assess data