Journal of the American Medical Informatics Association, 25(10), 2018, 1331–1338
doi: 10.1093/jamia/ocy093
Advance Access Publication Date: 30 July 2018
Research and Applications

Research and Applications

Web services for data warehouses: OMOP and PCORnet on i2b2

Jeffrey G Klann,[1,2,3] Lori C Phillips,[1] Christopher Herrick,[1] Matthew AH Joss,[1] Kavishwar B Wagholikar,[1,3] and Shawn N Murphy[1,2,4]

[1]Research Information Science and Computing, Partners Healthcare, Boston, Massachusetts, USA, [2]Harvard Medical School, Boston, Massachusetts, USA, [3]Department of Medicine, Massachusetts General Hospital, Boston, Massachusetts, USA and [4]Department of Neurology, Massachusetts General Hospital, Boston, Massachusetts, USA

Corresponding Author: Jeffrey G Klann, PhD, MGH Laboratory of Computer Science, 50 Staniford St., Suite 750, Boston, MA 02114, USA (jeff.klann@mgh.harvard.edu)

Received 7 January 2018; Revised 25 April 2018; Editorial Decision 20 June 2018; Accepted 28 June 2018

ABSTRACT

Objective: Healthcare organizations use research data models supported by projects and tools that interest them, which often means organizations must support the same data in multiple models. The healthcare research ecosystem would benefit if tools and projects could be adopted independently from the underlying data model. Here, we introduce the concept of a reusable application programming interface (API) for healthcare and show that the i2b2 API can be adapted to support diverse patient-centric data models.

Materials and Methods: We develop methodology for extending i2b2's pre-existing API to query additional data models, using i2b2's recent "multi-fact-table querying" feature. Our method involves developing data-model-specific i2b2 ontologies and mapping these to query non-standard table structure.

Results: We implement this methodology to query OMOP and PCORnet models, which we validate with the i2b2 query tool. We implement the entire PCORnet data model and a five-domain subset of the OMOP model. We also demonstrate that additional, ancillary data model columns can be modeled and queried as i2b2 "modifiers."

Discussion: i2b2's REST API can be used to query multiple healthcare data models, enabling shared tooling to have a choice of backend data stores. This enables separation between data model and software tooling for some of the more popular open analytic data models in healthcare.

Conclusion: This methodology immediately allows querying OMOP and PCORnet using the i2b2 API. It is released as an open-source set of Docker images, and also on the i2b2 community wiki.

Key words: medical informatics, data integration, data models, ontology-driven data representation, Patient Centered Outcomes Research Institute, Observational Health Data Sciences and Informatics, Informatics for Integrating Biology and the Bedside

BACKGROUND AND SIGNIFICANCE

Healthcare data models

A growing number of initiatives at several levels of scale are utilizing the vast quantity of information collected routinely by electronic health record (EHR) systems. Each initiative requires data be stored in its particular data model to support its shared analytical tools.

Popular models and major examples of use include:

- *Observational Medical Outcomes Partnership (OMOP)*. OMOP was a public-private partnership designed to develop methods and a data model to analyze observational healthcare data. The data model has been adopted by the Observational Health Data Sciences and Informatics (OHDSI) Consortium, a diverse, multi-stakeholder collaboration dedicated to providing robust analytical tools for research and quality improvement.[1,2] It has a large developer community and is presently used at approximately 90 sites worldwide.[3] It is the required data format of the AllOfUs Research Cohort, the massive federal undertaking to collect genotypic and phenotypic information on one million persons, which will increase its uptake.[4]

- *PCORnet Common Data Model (CDM)*. This CDM is intended to support the Patient Centered Outcomes Research Institute (PCORI)'s national network, PCORnet, and thus it is required for participation. This allows data partners to respond to SQL and SAS queries generated by PCORnet's Coordinating Center. PCORnet is a collection of data research networks that presently span almost 80 clinical sites nationwide to perform large-scale comparative effectiveness research.[5,6]
- *Informatics for Integrating Biology and the Bedside (i2b2)*. i2b2 is an open-source clinical data warehousing and analytics platform originally funded by the National Institutes of Health.[7] It is used at over 200 sites worldwide, including several PCORnet networks and the National Center for Advancing Translational Sciences (NCATS) Accrual to Clinical Trials (ACT) network.[8,9] Data in i2b2 can be queried by a well-honed cohort query tool with numerous analytics plugins that operate from RESTful Web Services.[10]

Each of these models is a giant step forward from the vast array of non-standard data repositories. OMOP and PCORnet offer robust CDMs for representing and analyzing EHR data. i2b2 uses a denormalized schema that has the flexibility to represent non-standard and local data.

Organizations must support multiple standard models based on the tools they plan to use and the projects they hope to be involved in. Therefore, much effort is being put into converting data into these various models, including our own previous work.[11] This requires maintaining multiple complex custom extract, transform, and load (ETL) processes.

The ecosystem of healthcare research would be improved if tools could be enabled to access data across models. Organizations would not be forced into a particular standardized data model based on the tools they need and could focus on the approaches that fit their data best. For example, our organization increasingly uses OMOP for analytics of typical EHR domains and i2b2 to capture local data that does not map to standard codes.

Application programming interfaces

For many years, computer technology has made use of application programming interfaces (APIs) to provide transparency to an underlying implementation. An API is a *lingua franca* used to communicate with software components. This is similar to Python or Java, which provide a common computable language on all platforms. In this age of the Internet, APIs are used to enable interoperability by exposing specific functionality in a standard way. The most popular Internet APIs are called "REST APIs."

Perhaps the most visible example of this is OAuth, the open standard for access delegation. Anyone who has logged into a website with the "Connect via Facebook" or "Connect via Google" button has used OAuth. Facebook and Google both support the common OAuth API, so that any site can provide its access delegation services to users.

OBJECTIVE

Unique among health-data CDMs, i2b2 provides a comprehensive data API that can be used for finding cohorts or retrieving individual patient information.[12] The API has been used to power not just i2b2's well-known graphical query tool, but many other projects emphasizing data interoperability and analytics. Some of these include SMART apps, HQMF queries, FHIR, Continuity of Care Document import, and a host of analytical plugins.[13–16]

Although this API is for the most part agnostic to the data it represents, at present, it has been implemented exclusively on the i2b2 data model. Therefore, up to this point, data must be transformed into it to utilize its features.

Here, we hypothesize that popular data models representing patient-centric data, including OMOP and PCORNet, can supported by the extant i2b2 API. Using the latest update to the i2b2 software, we develop a methodology for extending i2b2's preexisting REST API infrastructure to query additional data models. We utilize this methodology to develop a setup that can query the OMOP and PCORnet data models, which we validate with the i2b2 query tool. This architechture is outlined in Figure 1.

This will allow interested institutions to leverage the i2b2 API and tools while using the underlying data models best aligned with sites' other research goals.

MATERIALS AND METHODS

Healthcare data model approaches

i2b2's data model is a "star-schema." Its defining characteristic is one large "fact" table containing individual atomic observations. This is a narrow, long table with many rows per patient. Ancillary "dimension" tables provide additional context about, eg, the patient and encounter. Local implementations develop concept hierarchies (called "ontologies") that provide a window into the imported data. This ascribes metadata to the fact, such as "Cerebral Infarction" is ICD-10 code I63 and could have local code I10: I63. Import of new types of data elements can be done directly into the fact and dimension tables, and the ontology can be modified to make these data accessible to researchers.

Additional details about each data element (eg, primary vs. secondary diagnosis) are stored as modifier codes, which are also ascribed meaning through the ontology table. The fact table can store one modifier per row. Additional modifiers are added by duplicating the row in the fact table and changing only the modifier code. The i2b2 system knows this duplicate row is providing extra context on the original fact. Although, in theory, this can vastly expand the size of the fact table, in practice, only a few modifier types are available, which keeps the table computationally tractable.

Most other data models in medical informatics are designed with a normalized, column-oriented database structure, with many tables specialized for specific data domains, linked together by patient identifier and encounter number. This is true of PCORnet and OMOP.

PCORnet CDM's current release (v3.1) contains 15 tables, each corresponding to a clinical domain (eg, diagnoses, vitals, procedures, etc). The tables are wide, with many columns including both table keys (patient identifier, encounter identifier, etc) and additional details about each data element (eg, primary diagnosis flag).

OMOP's schema is more complicated, with 39 tables. Like PCORnet, it contains many domain-specific data tables. Unlike PCORnet, the domain tables consist of both raw data tables (eg, drug_exposure and visit_occurrence) and tables of derived values for specific analytical purposes (eg, drug_era and visit_cost). Similar to i2b2, OMOP provides metadata tables providing information on

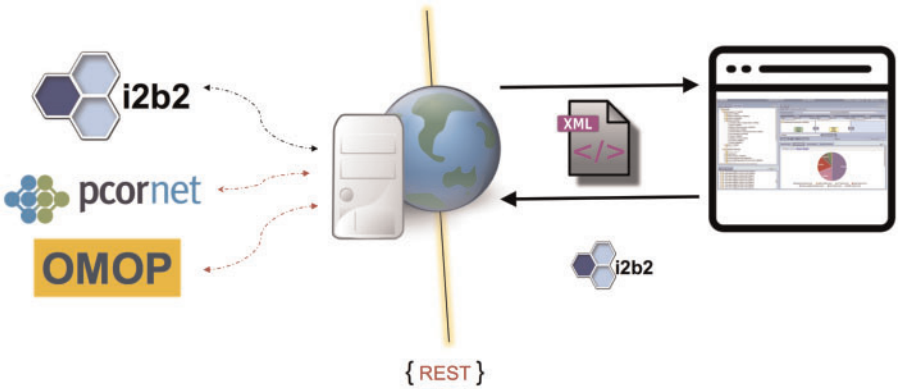


Figure 1. Web services on data warehouses. Shown here: the i2b2 database uses i2b2 XML REST services to communicate with the query tool. The bottom two arrows on the left show hypothetical connections to PCORnet and OMOP.

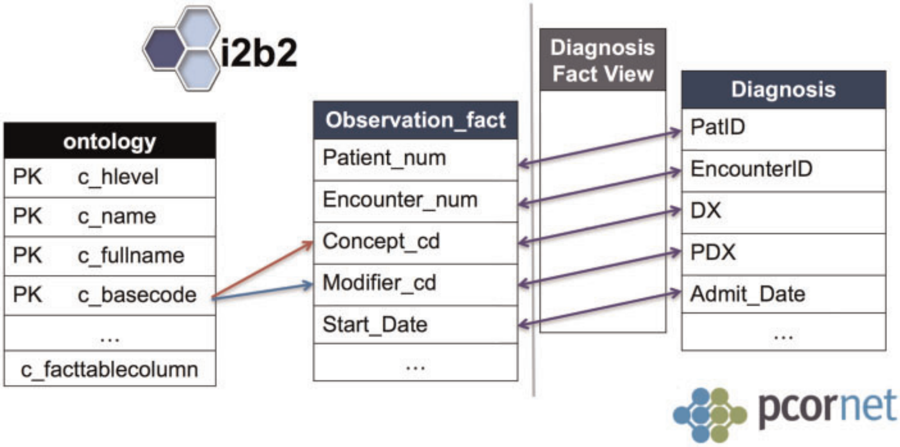


Figure 2. Showing the linkage between i2b2 ontology and non-i2b2 data models.

terminology and concept relationships. Unlike i2b2, however, this terminology is standardized and not modifiable at each site.

Representing new data models in the i2b2 API

The i2b2 API was originally an instantiation of its data model, so it can query and retrieve data stored in i2b2 format. The query language supports logical operations on i2b2 ontology elements and a variety of advanced data constraints. Results are returned as aggregate reports or patient-level data (see next section). i2b2 ships with a graphical query builder, allowing all levels of user expertise to use this API.

The key methodological insight in this work is that relational data models can be modeled as *a star-schema with multiple fact tables* without changing the underlying data. By doing this, the i2b2 API can directly query other patient-centric data models.

Our method for supporting a relational data model in the i2b2 API consists of the following steps:

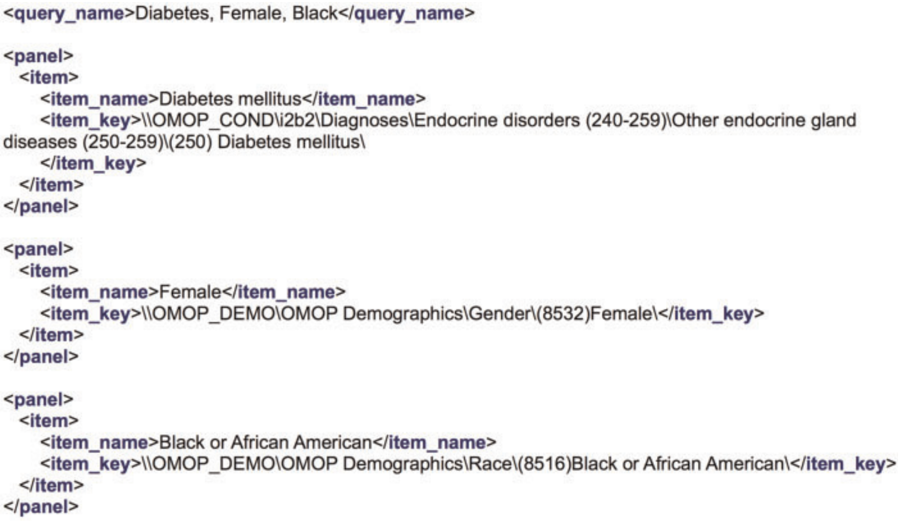
1. Install i2b2 1.7.09c, which supports specifying multiple fact tables in the ontology.
2. Create a star-schema database view of the target relational data model.
3. Develop an information model (ontology) that describes every possible fact in the desired relational data model, specifying the proper target fact table.

With this setup, the i2b2 API (and consequently all i2b2 tools) will function directly with a different underlying data model. Figure 2 visualizes these steps.

The *star-schema view* is a relational data model mapped into a star-schema with multiple fact tables. Generally, the patient dimension and visit dimension are represented by a corresponding patient table and encounter table in the data model. For the remaining domain tables, if the table can be formulated to have a primary key consisting of an encounter ID, patient ID, start and end dates, and concept ID (fact) column (eg, ICD diagnosis in a diagnosis table or RxNorm code in a medication table), then the table can be represented as a fact table.

A database view can be used to create a constant-time complexity column mapping of each source domain table into the i2b2 star-schema format. An example is shown in Figure 2. The database view can also support modifiers. As multiple modifiers are added using additional fact table rows, mapping modifiers involves "unpivoting" the source table, so that each modifier column is denormalized into another copy of the fact table row in the view. SQL provides highly efficient operations for this. Microsoft SQLServer offers a CROSS APPLY VALUES operation that will unpivot in constant time.[17] PostgreSQL does the same via an UNNEST(array[]) operation.[18] Oracle offers UNPIVOT.

Although there are many steps in this methodology, it introduces a trivial amount of computational complexity. The SQL views add



```
<query_name>Diabetes, Female, Black</query_name>

<panel>
  <item>
     <item_name>Diabetes mellitus</item_name>
     <item_key>\\OMOP_COND\i2b2\Diagnoses\Endocrine disorders (240-259)\Other endocrine gland
diseases (250-259)\(250) Diabetes mellitus\
     </item_key>
  </item>
</panel>

<panel>
  <item>
     <item_name>Female</item_name>
     <item_key>\\OMOP_DEMO\OMOP Demographics\Gender\(8532)Female\</item_key>
  </item>
</panel>

<panel>
  <item>
     <item_name>Black or African American</item_name>
     <item_key>\\OMOP_DEMO\OMOP Demographics\Race\(8516)Black or African American\</item_key>
  </item>
</panel>
```

Figure 3. A simplified query XML for the query in Figure 4, showing a query for all black, female patients with diabetes mellitus. The "item key" provides the unique ontology link that identifies each data element in the query.

only milliseconds to the query time. The column mappings use the indexes and other optimizations of the underlying tables. The column-to-row mappings (to implement modifiers) use highly efficient database-specific operations.

To give semantic meaning to the star-schema view, we utilize the flexible i2b2 ontology system, which is used to define the queryable terminology space for a particular i2b2 project. We call an i2b2 ontology that represents the possible concepts in a data model, the underlying information model.

We have previously defined and maintain an i2b2 information model for PCORnet CDM.[19] This presently supports PCORnet CDM 3.1 and has wide adoption among i2b2-based sites in the network.

An ontology item is defined by its *pathname*, which creates a file system-like hierarchy such as "\Diagnosis\ICD9\Endocrine Disorders\Diabetes Mellitus." Many of the other metadata fields in the ontology can be customized for particular use cases without sacrificing semantic integrity. We have previously utilized this flexibility, using the PCORnet ontology to be modified to query data with *non-standard codes in i2b2 tables.*[11] Here, we enable the subtly different case of querying *standardized codes in non-i2b2 tables*.

We employ a special feature in the ontology system that allows an implementer to specify the *column name, data type*, and (in i2b2 1.7.09) *table name* where data is stored for each item in the ontology. This feature, especially the addition of table name, enables the existing ontology system to query these star-schema views, because non-i2b2 table structure can be mapped into an i2b2 ontology.

Using the i2b2 API

The i2b2 API provides an extremely robust cohort-finding language (called SetFinder) and an interface to retrieve detailed patient data (called Patient Data Objects, or PDOs).

SetFinder retrieves a list of patient pseudoidentifiers and/or count of patients with optional stratifications. SetFinder queries support a variety of preliminary research, and capability has been demonstrated for cohort identification, phenotyping, and quality reporting.

SetFinder has two overall query approaches. The basic approach consists of one or more user-defined "panels" of ontology terms and modifiers combined through logical ANDs, ORs, and NOTs. An example of a SetFinder XML query is shown in Figure 3. Panels and terms can also be constrained by:

- Dates
- Minimum # of occurrences per patient record
- Value ranges, such as laboratory values, where applicable

A temporal-query SetFinder is also available, in which sophisticated temporal relationships can be defined among panel groups (eg, "all patients between 18-34 who were prescribed a beta blocker for the first time at least two weeks after the first diagnosis of atrial fibrillation").

A PDO request can then be performed to retrieve patient data on the resulting cohort or another specified patient list. PDOs can contain any data defined by the ontology, and ontology keys are used to specify the data domains of interest.

Implementation and testing

We performed the steps in the Summary section above to implement OMOP and PCORnet on i2b2, and we tested the ability and accuracy of the i2b2 query tool to return all types of SetFinder queries on OMOP and PCORnet datasets. We tested PDO retrieval by running a sample app used for viewing a patient chart, through the i2b2-SMART-on-FHIR project.[13] We also tested query performance to verify that this machinery did not introduce significant computational complexity.

RESULTS

I2b2 on OMOP

We first created an OMOP v5.1 database and imported the 1000-patient synthetic patient dataset (synPUF), developed by CMS and supplied in OMOP format by the OHDSI organization. Next, we developed an OMOP information model for i2b2, guided by the OMOP domains utilized by synPUF. The domains covered by the ontology and the terminologies supported are summarized in Table 1. We built these terminology trees using a pre-existing tool to generate i2b2 ontologies from BioPortal,[20] or from other previous work.[11] At present, the "modifier" columns in OMOP (eg, refills,

Table 1. i2b2 ontologies to OMOP tables

i2b2 ontology tree	Provided I2b2 terminologies	Standardized OMOP terminologies	Target OMOP tables
Diagnosis	ICD-9, SNOMED	SNOMED	Condition occurrence, Measurement,Procedure occurrence, Observation
Procedures	HCPCS, ICD-9, SNOMED, ICD-10	*Same as i2b2 terminologies*	Procedure occurrence, Device exposure, Drug exposure, Observation
Medications	RxNorm, NDC	RxNorm	Drug exposure
Labs	LOINC, SNOMED test findings	LOINC*, SNOMED	Measurement
Demographics	Age, ethnicity, gender, race	Custom value set	Person

*LOINC codes are not used in the SynPUF data, but we did implement support for these.

Table 2. OMOP tables mapped as fact tables into i2b2

OMOP table	Concept code	Start date	End date	Value
Condition occurrence	Condition concept ID	Condition start date	Condition end date	
Drug exposure	Drug concept ID	Drug exposure start date	Drug exposure start date	
Procedure occurrence	Procedure concept ID	Procedure date	–	
Measurement	Measurement concept ID	Measurement date	–	Numeric value from value_as_number
Observation	Observation concept ID	Observation date	–	Value_as_string and value_as_number

Table 3. i2b2 ontologies to PCORnet tables

i2b2 ontology tree	i2b2 terminologies	Target PCORnet table(s)
Diagnoses	ICD-9, ICD-10	Diagnosis, condition
Procedures	HCPCS, CPT, ICD-9, ICD-10	Procedure
Medications	RxNorm, NDC	Prescribing, dispensing
Labs	LOINC	LabResult
Demographics	Age, race, ethnicity, gender, sex, sexual orientation	Demographic
Vitals	Height, weight, blood pressure, smoking status	Vitals
Enrollment	n/a	Enrollment

quantity, etc.) have not been tested, because they are not present in the synPUF data.

In order to make our information model able to query OMOP datasets:

1. We created fact views for the corresponding OMOP tables. The Person table maps to the patient dimension, the Visit table maps to the encounter dimension, and the remaining five ontology domains map to eight OMOP tables.
2. We replaced the underlying codes in the ontology with its equivalent standardized OMOP number from the OMOP concept dictionary. In cases in which the terminology is considered nonstandard by OMOP, we used the OMOP mapping to a standardized equivalent here.
3. We assigned every element in the ontology to the target fact view *table name* specified by the OMOP concept dictionary.

Table 2 provides more details on the fact views (step 1 above), showing how columns from OMOP are mapped to i2b2. Table 1 summarizes the results of the remaining steps, showing how each i2b2 ontology tree is assigned a target OMOP terminology and table. Generally, there is a one-to-one mapping between categories and OMOP tables; however, procedures and diagnoses map to several OMOP tables. For example, a HCPCS code for a catheter is found in the device exposure table, and a diagnosis code for abnormal glucose tolerance is assigned to the measurement table. The domain design is articulated on the OHDSI wiki.[21] As mentioned in step 3, these ontologies' elements are assigned a target table using the ontology "target table-name" feature.

We connected this setup to an i2b2 project, pointed the i2b2 query tool to the project, and verified our ability to perform queries, across all ontology domains. This verification is detailed further below, in the subsection Validating Correctness.

We have released this toolset on the i2b2-on-OMOP wiki page.[22]

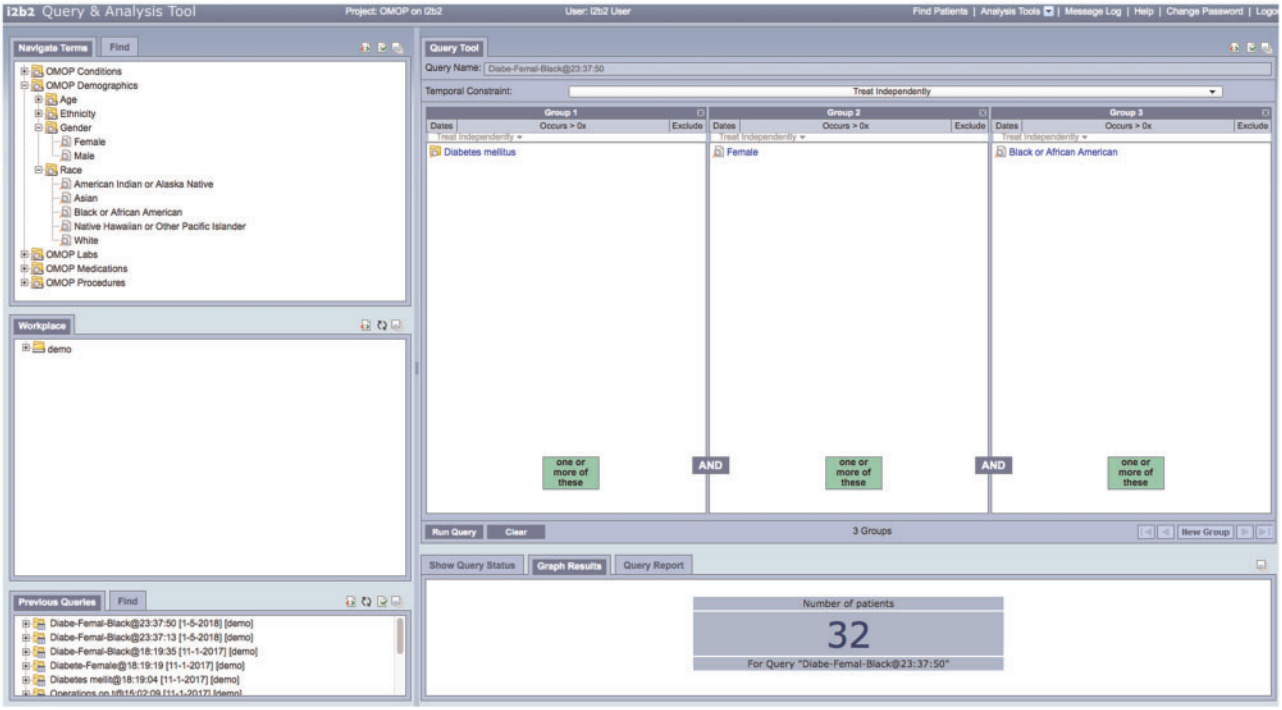
i2b2 on PCORnet

We adopted the PCORNet Version 3.1 information model for i2b2.[11] It can be found at the GitHub site for our network, the Accessible Research Commons for Health (ARCH).[19] Because synPUF data in PCORnet CDM format is not available, we used a previously developed PCORnet version of the 133-synthetic-patient i2b2 dataset distributed with i2b2. The domains covered by the PCORnet ontology and the terminologies supported are summarized in Table 3.

As seen in Table 3, in some cases we represent multiple PCORnet tables as a single ontology tree (eg, Diagnosis and Condition are both queryable through the Diagnosis tree). This is accomplished through a modifier flag called "Data Type," which is used to select whether the data are conditions or diagnoses. This setup eliminates the need for two essentially equivalent ontology trees. It creates a small complexity in creating the fact views, in that one fact view

Table 4. PCORnet tables mapped as fact tables into i2b2

PCORnet Table	i2b2 Domain	Concept code	Start date	End date	Modifiers	Value
Condition	Diagnosis	Condition_type +condition	Report date	Resolve date	–	
Diagnosis	Diagnosis	DX type + DX	Admit date	–	Primary DX, DX source	
Dispensing	Medication	NDC	Dispense date	–		
Enrollment	Enrollment	Enrollment basis	Enrollment start	Enrollment end	Chart	
Labs	Labs	LOINC code	Specimen date	Result date		Numeric value from result_num
Prescribing	Medication	RxNorm CUI	RX start date	RX end date	RX frequency	
Procedure	Procedure	PX type + PX	Admit date	PX date	–	



Query Name: Diabe-Femal-Black@23:37:50

Temporal Constraint: Treat Independently

Group 1: Diabetes mellitus — Treat Independently

Group 2: Female — Treat Independently

Group 3: Black or African American — Treat Independently

3 Groups

Number of patients
32
For Query "Diabe-Femal-Black@23:37:50"

Diabe-Femal-Black@23:37:50 [1-5-2018] [demo]
Diabe-Femal-Black@23:37:13 [1-5-2018] [demo]
Diabe-Femal-Black@18:19:35 [11-1-2017] [demo]
Diabete-Female@18:19:19 [11-1-2017] [demo]
Diabetes melli@18:19:04 [11-1-2017] [demo]
Operations on t@15:02:09 [11-1-2017] [demo]

Figure 4. i2b2 querying OMOP's synPUF 1000-patient dataset. This query took 2.2 seconds.

must be created for the union of the two PCORnet tables, with the default modifier set differently for each source table. Note that this is different from OMOP in which eg, Diagnoses go into four tables. In OMOP, the target table is pre-determined by the OMOP vocabulary, rather than data driven. Thus, OMOP's target table is defined in the ontology, rather than the more complex view.

We next generated fact table views for the PCORnet CDM. The demographics table maps to the patient dimension, the encounter table maps to the visit dimension, and the remaining seven tables map to each of the five ontology domains, which are outlined in detail in Table 4. The table shows the main concept mapped for each fact view, as well as how the key date constraints in i2b2 (start and end date) were modeled for each table. Also shown are the modifier columns currently implemented. For this work, we implemented only key modifiers, and we demonstrated on the diagnosis tree that we can successfully map multiple modifiers.

Our initial release of this implementation can be found on our GitHub site.[23]

Validating correctness

For both PCORnet and OMOP, we used our sample database (synPUF data for 1000 patients on OMOP, synthetic demodata on PCORnet) and ran all major query types on all domains. A sample i2b2 query of OMOP data can be seen in Figure 4, with its underlying XML representation shown in Figure 3.

We verified that all of the following SetFinder query types are functional for both OMOP and PCORnet:

- Queries on every individual domain
- Modifiers (for the PCORnet modifiers shown in Table 4)
- Multi-panel, multi-domain queries
- Date constrained queries
- Occurs > x queries
- Value constrained queries
- Temporal queries

We also verified the results by comparing our query counts to unmodified i2b2 populated with the same data, for both data models. Each query returned exactly the same results on unmodified i2b2 vs. OMOP/PCORnet-on-i2b2.

Our ability to run value-constrained queries was limited in OMOP by the SynPUF data set, as it did not contain any value-based fact data. To test this, we created a small set of value-based measurement and observation lab data for five patients.

Finally, to test the retrieval of detailed patient data, we executed a proof-of-concept SMART app using the i2b2-FHIR cell.[13] We were able to successfully authenticate using the OAuth2.0 protocol to retrieve and display patient data using the i2b2 web services on the OMOP and PCORNet databases.

Validating performance

To verify that this methodology does not introduce significant computational complexity, we performed a performance test, comparing the i2b2-on-OMOP project[24] to an out-of-the-box i2b2 datamart. We loaded the OMOP SynPUF 5% v1.0.0 dataset of 99 210 synthetic patients as an i2b2-on-OMOP project.[24] For i2b2 comparison, we selected an i2b2 datamart at Partners Healthcare consisting of 74 648 patients and a similar quantity of total data.

We executed eight variants of a diabetes query on both systems, to test each clinical domain, date constraints, breakdowns, "occurs > x," and temporal constraints. (We were unable to include value constraints due to previously described SynPUF dataset limitations.)

i2b2 and i2b2-on-OMOP performed similarly for all queries. i2b2-on-OMOP returned results on an average (median) of 6.5 seconds, and i2b2 took an average (median) of 5.5 seconds. (Full details are available in the Supplementary Appendix.)

DISCUSSION

We have developed a methodology to utilize i2b2's REST API on multiple healthcare data models, thus enabling tooling that is based on an API rather than a particular choice of backend data stores. This will allow tools to interoperate with datasets in a variety of standard data models without maintaining a separate ETL process for each model. Furthermore, it could support analytics transparently on combinations of data models, such as using an OMOP data model for standard EHR domains and an i2b2 data model for data domains that are not yet incorporated into OMOP, such as genomics.

We implemented proof-of-concept implementations that allow querying of OMOP and PCORnet CDM databases through the i2b2 query tool, and we tested these on a 1000-synthetic-patient dataset from CMS and a 133-synthetic-patient dataset created from i2b2, respectively. We were able to successfully query against OMOP and PCORnet fact tables and produce correct query results. Also, we performed a speed test that showed comparably fast query speed on a 100k-patient dataset in both i2b2-on-OMOP and i2b2, indicating that our mapping machinery does not cause appreciable performance degradation. This, along with our complexity analysis, gives us reason to believe that the approach will also be performant in a large-scale, optimized, production setting.

We have deployed this tooling as a set of Docker images, based on the latest i2b2 Docker images in DockerHub. These images include i2b2 1.7.09c and a Postgres database with both the i2b2-on-OMOP and i2b2-on-PCORnet projects with their respective demo data.[25,26]

LIMITATIONS

A major caveat, as is often true in informatics, is terminology. While i2b2 provides a means to represent any healthcare data model as standard XML messages and standard ontologies, this does not necessarily mean the same XML query will work on all models, because the query definition is dependent on the underlying ontologies. While i2b2 running on PCORnet will query ICD, i2b2 running on OMOP might query SNOMED. Even tools built on a common API must consider multiple standard vocabularies.

Another limitation is that we have not yet implemented many "modifiers" in our ontologies/views, which will allow users to query the additional columns in OMOP and PCORnet. Our methodology allows the remaining modifiers to be implemented, but complexities always occur during actual implementation. For example, i2b2 was designed for a large database of core facts with a small number of modifiers, in response to actual data availability in Enterprise Data Warehouses. However, because of the low complexity cost in adding additional columns, table-based models support many potential modifiers. Although we have shown that many columns-as-modifiers can be added to the star-schema view without increasing computational complexity, nonetheless, the fact table view potentially doubles in size for each modifier added. Therefore, one might expect some degree of query performance degradation when enough modifiers are added.

A minor limitation is that synPUF data are unavailable for PCORnet CDM, so our OMOP and PCORnet implementations were tested on different data sets. For validation purposes, this is not an issue: we confirmed query correctness by comparing results to the same data in unmodified i2b2, so common test data between i2b2 and OMOP are not needed. However, this does preclude full, potentially interesting comparisons of data representation and query speed between CDM implementations.

CONCLUSION

This methodology of applying the i2b2 API to many data models provides a means for separation between data model and tooling. It could allow implementers to select the data models of greatest utility to their institution without needing to change with every funded project (which seems to be the current vex of informatics), and in turn, it could allow projects to query clinical data without requiring sites to adopt a particular data model.

FUNDING

This work was supported by the Patient-Centered Outcomes Research Institute (PCORI), award number CDRN-1306-04608; NIH NHGRI grant number U54 HG007963; NIH NHGRI grant number RO1 HG009174; and the National Library of Medicine (NLM), grant number 5R00LM011575.

PCORI DISCLAIMER

The statements presented in this publication are solely the responsibility of the author(s) and do not necessarily represent the views of the Patient-Centered Outcomes Research Institute (PCORI), its Board of Governors or Methodology Committee, or other participants in PCORnet.

Conflict of interest statement. The authors declare they have no competing interests.

CONTRIBUTORS

JGK wrote the majority of the paper, developed the PCORnet ontology, led the implementation of i2b2-on-PCORnet, and contributed the final Docker images. LCP developed the software changes to i2b2 and developed the implementation of i2b2-on-OMOP, including building ontology. CH was responsible for the overall design/architecture changes for new data models running on i2b2. MAJ contributed to both the i2b2-on-PCORnet implementation and the speed testing. KW edited the paper and led dockerizing the tools. SNM formulated the idea, is the director of i2b2, and leads strategic direction.

SUPPLEMENTARY MATERIAL

Supplementary material is available at *Journal of the American Medical Informatics Association* online.

ACKNOWLEDGMENTS

Thanks to the entire i2b2 development team and all the members of the i2b2, OMOP, and PCORnet communities who have made this work possible.

REFERENCES

1. OHDSI | Observational Health Data Sciences and Informatics. http://www.ohdsi.org/ Accessed July 29, 2015.
2. Hripcsak G, Duke JD, Shah NH, *et al*. Observational Health Data Sciences and Informatics (OHDSI): opportunities for observational researchers. *Stud Health Technol Inform* 2015; 216: 574–8.
3. OHDSI. resources: data_network. Observational Health Data Sciences and Informatics. http://www.ohdsi.org/web/wiki/doku.php?id=resources:data_network Accessed April 18, 2018.
4. All of Us Research Program. National Institutes of Health (NIH). https://allofus.nih.gov/ Accessed March 6, 2017.
5. Collins FS, Hudson KL, Briggs JP, *et al*. PCORnet: turning a dream into reality. *J Am Med Inform Assoc* 2014; 21 (4): 576–7.
6. PCORnet Common Data Model (CDM). PCORnet. http://www.pcornet.org/pcornet-common-data-model/ Accessed March 6, 2017.
7. Murphy SN, Weber G, Mendis M, *et al*. Serving the enterprise and beyond with Informatics for Integrating Biology and the Bedside (i2b2). *J Am Med Inform Assoc* 2010; 17 (2): 124–30.
8. McMurry AJ, Murphy SN, MacFadden D, *et al*. SHRINE: enabling nationally scalable multi-site disease studies. *PLoS One* 2013; 8 (3): e55811.
9. CTSA Consortium Tackling Clinical Trial Recruitment Roadblocks. National Center for Advancing Translational Sciences. 2015. https://ncats.nih.gov/pubs/features/ctsa-act Accessed September 21, 2016.
10. i2b2 Community Plugins. GitHub. https://github.com/i2b2plugins Accessed January 6, 2018.
11. Klann JG, Abend A, Raghavan VA, *et al*. Data interchange using i2b2. *J Am Med Inform Assoc* 2016; 23 (5): 909–15.
12. Murphy SN, Mendis M, Hackett K, *et al*. Architecture of the Open-source Clinical Research Chart from Informatics for Integrating Biology and the Bedside. *AMIA Annu Symp Proc* 2007; 2007: 548–52.
13. Wagholikar KB, Mandel JC, Klann JG, *et al*. SMART-on-FHIR implemented over i2b2. *J Am Med Inform Assoc* 2017; 24 (2): 398.
14. Klann JG, Murphy SN. Computing health quality measures using Informatics for Integrating Biology and the Bedside. *J Med Internet Res* 2013; 15 (4): e75.
15. Klann JG, Mendis M, Phillips LC, *et al*. Taking advantage of continuity of care documents to populate a research repository. *J Am Med Inform Assoc* 2014; 22 (2): 370–9.
16. Wattanasin N, Porter A, Ubaha S, *et al*. Apps to display patient data, making SMART available in the i2b2 platform. *AMIA Annu Symp Proc* 2012; 2012: 960–9.
17. Kenneth Fisher. UNPIVOT a table using CROSS APPLY. SQL Studies. 2013. https://sqlstudies.com/2013/04/01/unpivot-a-table-using-cross-apply/ Accessed April 23, 2018.
18. unpivot and PostgreSQL - Stack Overflow. https://stackoverflow.com/questions/1128737/unpivot-and-postgresql Accessed April 23, 2018.
19. Accessible Research Commons for Health Github. GitHub. https://github.com/ARCH-commons Accessed March 6, 2017.
20. Phillips L. NCBO Extraction Tool Version 2.0. NCBO Ontology Tools for i2b2. https://community.i2b2.org/wiki/display/NCBO/NCBO+Extraction+Tool+version+2.0 Accessed December 4, 2014.
21. CommonDataModel: Definition and DDLs for the OMOP Common Data Model (CDM). Observational Health Data Sciences and Informatics 2017. https://github.com/OHDSI/CommonDataModel Accessed December 22, 2017.
22. i2b2 on OMOP - Release 1.7.09. https://community.i2b2.org/wiki/display/OMOP/Release+1.7.09 Accessed November 15, 2017.
23. Klann JG, Joss M. i2b2-on-PCORnet: Scripts and Documentation for Implementing. Accessible Research Commons for Health 2017. https://github.com/ARCH-commons/arch-utils/tree/master/i2b2_on_PCORnet Accessed January 4, 2018.
24. ETL-CMS: Workproducts to ETL CMS datasets into OMOP Common Data Model. Observational Health Data Sciences and Informatics 2017. https://github.com/OHDSI/ETL-CMS Accessed January 7, 2018.
25. pcori/omop i2b2 demo. https://github.com/waghsk/i2b2-quickstart/wiki/pcori-demo Accessed November 15, 2017.
26. Install dockerized i2b2 on PCORnet/OMOP. https://github.com/ARCH-commons/arch-utils/wiki/Install-dockerized-i2b2-on-PCORnet Accessed November 15, 2017.