

# i2b2 implemented over SMART-on-FHIR

Nicolas Paris, PhD Student<sup>1, 2</sup>, Firstname B. Lastname, Degrees<sup>2</sup>

<sup>1</sup>WIND-DSI, AP-HP, Paris, France; <sup>2</sup>CNRS, LIMSI, Orsay, France; <sup>3</sup>INSERM, UMR\_S 1142, LIMICS, Paris, France;

## Abstract

*Integrating Biology and the Bedside (i2b2) is the de-facto open-source medical tool for cohort discovery. Fast Healthcare Interoperability Resources (FHIR) is a new standard for exchanging health care information electronically. Substitutable Modular third-party Applications (SMART) defines the SMART-on-FHIR specification on how applications shall interface with EHR through FHIR. Related work made possible to produce FHIR from an i2b2 instance or i2b2 to store FHIR datasets. In this paper, we extend i2b2 to search remotely into one or multiple SMART-on-FHIR API. This enables federation of queries, security, terminology mapping, and also bridges the gap between i2b2 and modern big-data technologies.*

## Introduction

Learning Health Systems aim to maximize the potential of large-scale, harmonized data from variable, quickly-developing digital sources including Electronic Health Records (EHRs) emerging as a powerful tool to facilitate discoveries that can improve health. Data heterogeneity is one of the critical problems in analyzing, reusing, sharing or linking datasets. With the development of platforms enabling the linking and federation of phenome, genome and exposome data across sites in US (14), Europe (5,6) or at international scale (7) a key challenge is to define harmonized access to heterogeneous EHR-based data.

In the domain of patient care, several large-scale efforts have been underway for over a decade with the goal of specifying both the structure and the semantics of patient clinical information in a manner that enables computable semantic interoperability between diverse systems. Two major contributions to the interoperability of clinical information currently dominate internationally: i) the ISO EN 13606 is a generic and comprehensive representation for the interchange of EHR information between heterogeneous systems suited to the extraction, communication and/or mapping of longitudinal EHR data or of fine grained parts of an EHR. The openEHR Foundation maintains a more detailed model, catering for the widest set of use cases for patient level data; ii) the Health Level Seven (HL7) Fast Healthcare Interoperability Resources (FHIR), built on lessons from previous standards including the Reference Information Model (RIM) that became an ISO standard in 2003 and Clinical Document Architecture, designed to express a single clinical document as a message using HL7 version 3 RIM classes.

Both EN ISO 13606 and HL7 FHIR standards define the semantics of patient care data and clearly demonstrate the need for layers of semantic expressiveness including: i) generic reference information models of concepts and relationships (e.g. EN ISO 13606, openEHR Reference Model or FHIR model) each capable of binding terms from terminology models (e.g. SNOMED-CT, LOINC, etc.) and associated with a data type models such as ISO 21090; and ii) more detailed models (e.g. EN ISO 13606 or openEHR Archetypes/Templates or FHIR resources).

Although there is no consensus in the medical informatics community regarding a standard patient information model, HL7 FHIR specifications are gaining interest and show promise to mitigate the classic site-specific data mapping problem. FHIR specifies a RESTful application programming interface (API) to access resources. Several initiatives aim to facilitate the adoption of FHIR, including the Argonaut project (8), the Data Access Framework (9) and the Clinical Information Modeling Initiative (CIMI) launched in 2011, an international consortium of representing national bodies, Standards Development Organizations, healthcare organizations and vendors are building collaboratively a process and tools for constructing a single curated collection of shared implementable clinical information models that are free for use at no cost (10).

SMART Health IT is an open, standards based technology platform that enables innovators to create apps that seamlessly and securely run across the healthcare system. Using an electronic health record (EHR) system or data warehouse that supports the SMART standard, patients, doctors, and healthcare practitioners can draw on this library of

apps to improve clinical care, research, and public health (11). SMART success improve the user experience exactly the same major internet provide access to many application with a single authentication.

i2b2 is the de-facto open-source medical tool for cohort discovery and allows healthcare practitioners to easily query patient data to address research questions. Many initiatives have extended this primary goal with statistical analysis on place[ref], federating queries over multiple centers [shrine, insite, triketX], and even genomics analytics.[transmart, i2b2-transmart]. Its recent migration on github allows multiple developers to improve and extend the source code. I2b2 has been described to be used by more than 200 hospitals over the world. The tool is flexible and can support its own stars schema and ontology model, or exploit new information models e.g. PCORnet or OMOP common data model - without requiring changes to the underlying data .ETL processes feeding the databases are time, resource, maintenance and disk space consuming processes. The database layer can be PostgreSQL, Oracle or Microsoft SQL Server. Though ETL processes are still feasible these days, the emergence of high throughput healthcare data data and the Internet of Things requires the development of new approaches.

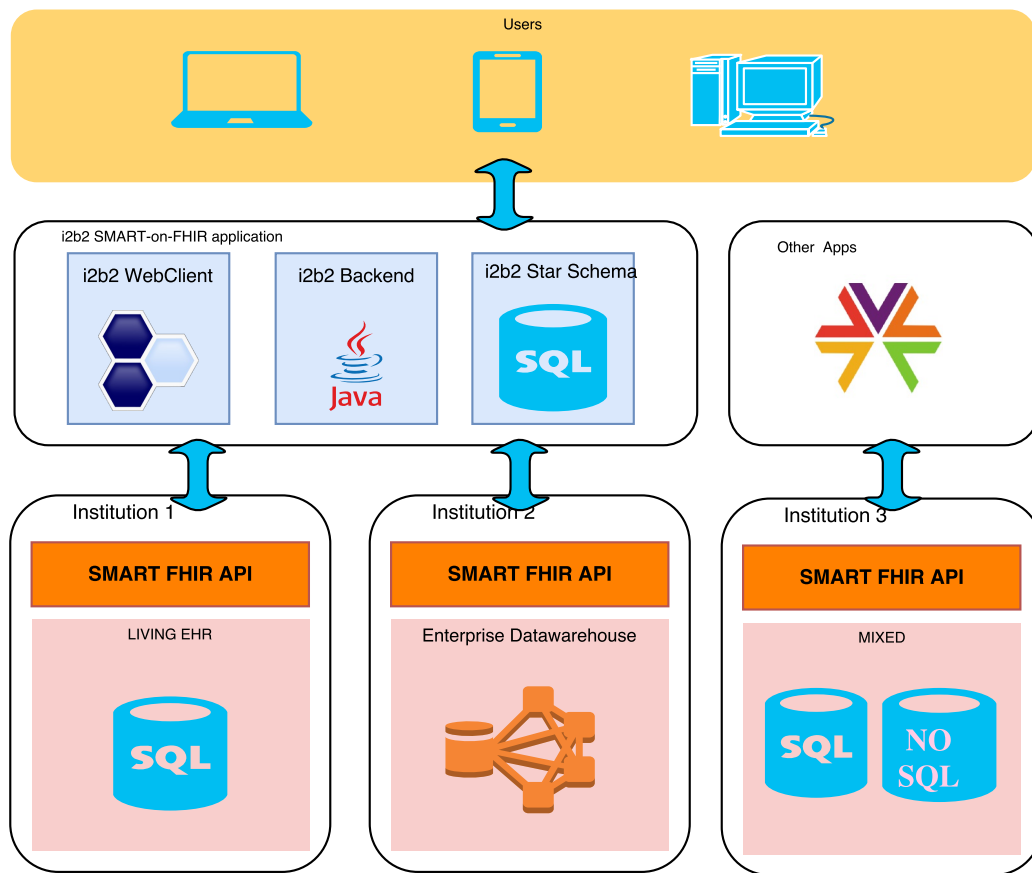
The objectives of this work is to bring the latest accomplishments of the FHIR community to i2b2. In particular, bring the flexibility, the extensibility, the standardisation, and the interoperability efforts to i2b2. This work describes a general interface between i2b2 and any type of clinical dataset by exploiting the FHIR search[ref], Terminology Mapping[ref] and SMART Oauth2 security[ref] specifications. The aim was not only to bridge the gap between patient care and research communities, but also to open to i2b2 new areas for better data types, security and interoperability management in the context of scalable solutions for cross border and cross domain networking of data.

## Methods

To meet the objectives, the existing i2b2 CRC cell code source is extended with code that meet the SMART-on-FHIR API specifications and the FHIR search API specifications. The Figure 4 shows the overall architecture and how the three-tier i2b2 application articulates with 3 remote institutions. The figure shows how i2b2 application gives access to users in a SMART-on-FHIR application context. In this context, users log one time in any SMART application or EHR, and get access to their specialized applications available. Moreover, the architecture allows to mix queries over multiples endpoints: zero to one i2b2 star schema and/or zero to many SMART-on-FHIR APIs.

The Figure 5 is a detailed UML. The scenario describes a user who query over an i2b2 instance with multiple remote FHIR-endpoints accesses. The user first logs-in with its personnal secrets informations, that are verified by the i2b2 project management cell (i2b2pm). The i2b2pm then asks and stores a Oauth2 credential to all the SMART authentication services with it's own i2b2 connection details (one global secret for the i2b2 application) dedicated for the user. The i2b2pm returns then an i2b2 project list, to let the user choose and access according its habilitation details defined into i2b2. The user builds and run a multiple panel query accross different medical domains to get back a patient cohort set. The i2b2crc will then loop the following steps over each panel and each SMART-on-FHIR API. The i2b2crc transforms the query according to the FHIR-search specifications and passes it with the credentials to the FHIR-API. The Oauth2 credential information are verified by the FHIR-API, and the query extended with the coding with synonyms defined in the terminology conceptMapping. The resulting query is then translated by the FHIR-API in the local database dialect to fetch the result. The result is transformed into a FHIR json bundle only containing the information needed (patient\_ids in this case). A parsing step extract the patient\_ids. They are mapped to an i2b2 unique identifier thanks to the existing i2b2 patient\_mapping features, to be then pushed into a CRC temporary table that integrates all the results. Once looping done, the i2b2crc applies the patients security steps to the CRC tmp table in order to only keep the patients that are available for the project selected by the user. The patient cohort set is finally returned to the user.

*FHIR-search:* FHIR search specifications describe how to communicate with a FHIR-API to get back a set of resources matching an HTTP query criteria. The present work exploits only the possibility to fetch one type of resource per query. This is sufficient because i2b2 traditionnal query search module (i2b2crc) allows combining multiple filters predicates processing each separately and then uses a deliberation step using temporary tables. The idea of the query builder extension, is to being able to replace the i2b2crc SQL queries acting over the star schema to fetch records identifiers (ID) with HTTP calls to a FHIR-API and then any database system behind. The HTTP calls enabled in this design are presented in Table ???. The first row is the general template used, and has an analogy with SQL syntax:



**Figure 1: Overall Diagram**

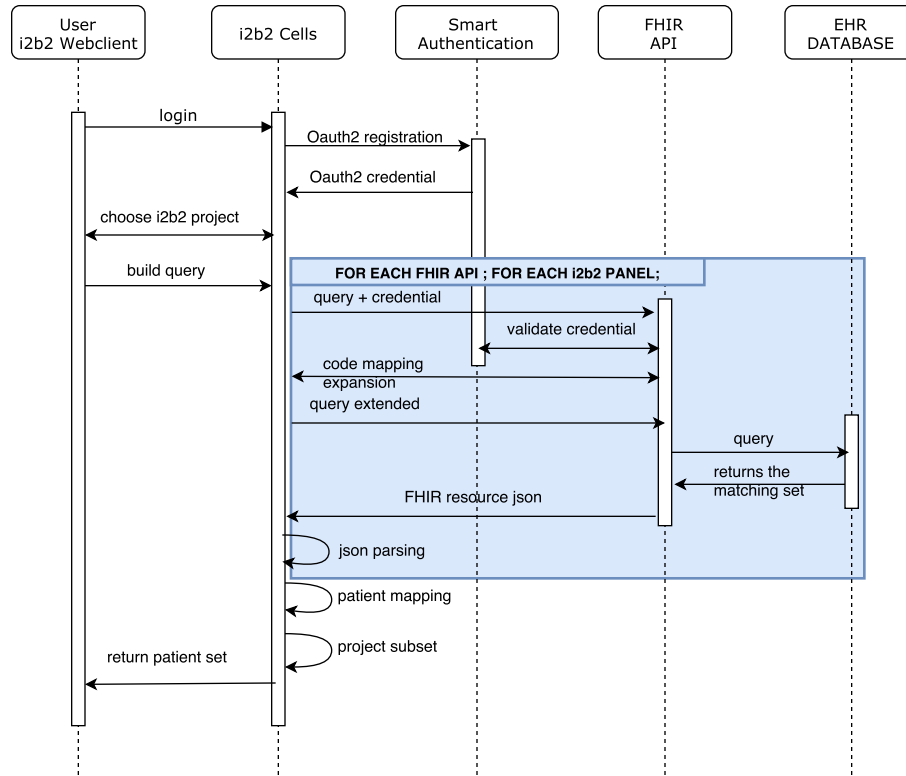
**SELECT:** The <elements> pattern lists the resource elements that are returned by the FHIR-API. Depending on the user choice, patient ID, encounter ID, instance ID or date are retrieved. The way to retrieve those information from a given resource is described into the i2b2 FHIR config YAML file Figure

**FROM:** The <Resource> pattern is supposed to be replaced by any existing FHIR standard resource, or any profiled resource (modification of the standard to meet the local institutions constraints).

**WHERE:** Both patterns <date.inf> and <date.sup> allows filtering the data based on date range. The <custom.filter> allows to combine a predefined pattern, such data status, or a user defined constraint by value query. The <codes> pattern can optionally contain a list of coding (eg:SNOMED, LOINC...).

*FHIR-mapping:* The second row of the Table ?? describe the template to enable the terminology mapping:

*Outcome measurement:* In order to test the FHIR DSTU3 resources compatibility coverage, the HAPI FHIR test server has been used has endpoint since it contains useful demo datasets for the 68 resources. The benchmark comparing traditionnal i2b2 and FHIR-i2b2 has been done with the same i2b2 observation\_fact table containing 140 Million records in a postgresql 9.6 instance. The first is based on a 1.7 i2b2 instance. The FHIR-i2b2 has been setup by implementing HAPI-FHIR server on top of the observation\_fact table into an apache tomcat 9 webserver, and accessed via a the FHIR-i2b2 prototype. The FHIR-i2b2 big-data benchmark has been setup by implementing HAPI-FHIR server on top of a MIMICiii table multiplied by 15, and stored in a apache HIVE2 table distributed over a 5 computer cluster in ORC format. All software used: i2b2, HAPI-FHIR, postgresql and apache Hive are open-source.



**Figure 2:** UML Sequence Diagram

ontology table columns	Description	Example
c_basecode	FHIR code_system / code pipe separated	FHIR:http://loinc.com 1234-5
c_facttable	Resource / Profile pipe separated	Observation ObservationAphp
c_metadataxml	An xml describing datatype (numeric, free text or enumerated) and measure units	cf: i2b2 documentation
c_concept_cd	an optionnal additional filter	active=true&status=final

**Table 1:** i2b2 ontology adapted for FHIR

## Results

*Implementation Status:* The design presented below is not yet fully implemented. To date, the query builder is able to query on both star schema and one remote FHIR endpoint simultaneously. Logical relation between selection criteria represented as multiple i2b2 webclient panels are also possible. The constitution of a patient\_set can be constraint by dates, by values and mesurement units and by one or multiple codes. The code expansion based on FHIR terminology mapping is also implemented. A living demo is deployed at <http://34.205.31.28/webclient/> and its screenshot presented in Figure 6. The panel 1 query searched into HAPI FHIR test server for patients with a set of loinc glucose codes having value lower than 100ml/dl in a year range from 1979 to 2015 and is mixed with the panel 2 searching for patients diagnosed related to circulatory system within the star schema. The resulting patient\_set is about 8 patients.

*Performances:* The performances have been benchmarked (Figure 4) versus a traditionnal i2b2 instance based on star schema with the same amount of data, and configuration (140 Million records). The histogram shows traditionnal i2b2 is 20 times faster than the i2b2-FHIR version. The difference can be explained by the additionnal steps involved: the fetched resultset is transformed into a json bundle, sent over the network and then parsed. The performance factor tends to decrease with number of patient matched. The second benchmark (Figure 5) experienced connecting to a

HTTP request	Description
GET <FHIR-BASE>/<Resource> ?_elements=<elements>&code=<codes> &date=gt<date_inf>&date=lt<date_sup> &<custom_filter>	Retrieves choosen <elements> from resources optionally matching a date range or/and a list of <codes> or/and a <custom_filter>
GET <FHIR-BASE>/ConceptMapping ?target-code=<codes>&target-system:in=<code-system>	Retrieves all codes that are mapped to <codes> & <code-system>

**Table 2:** Index of HTTP requests

```

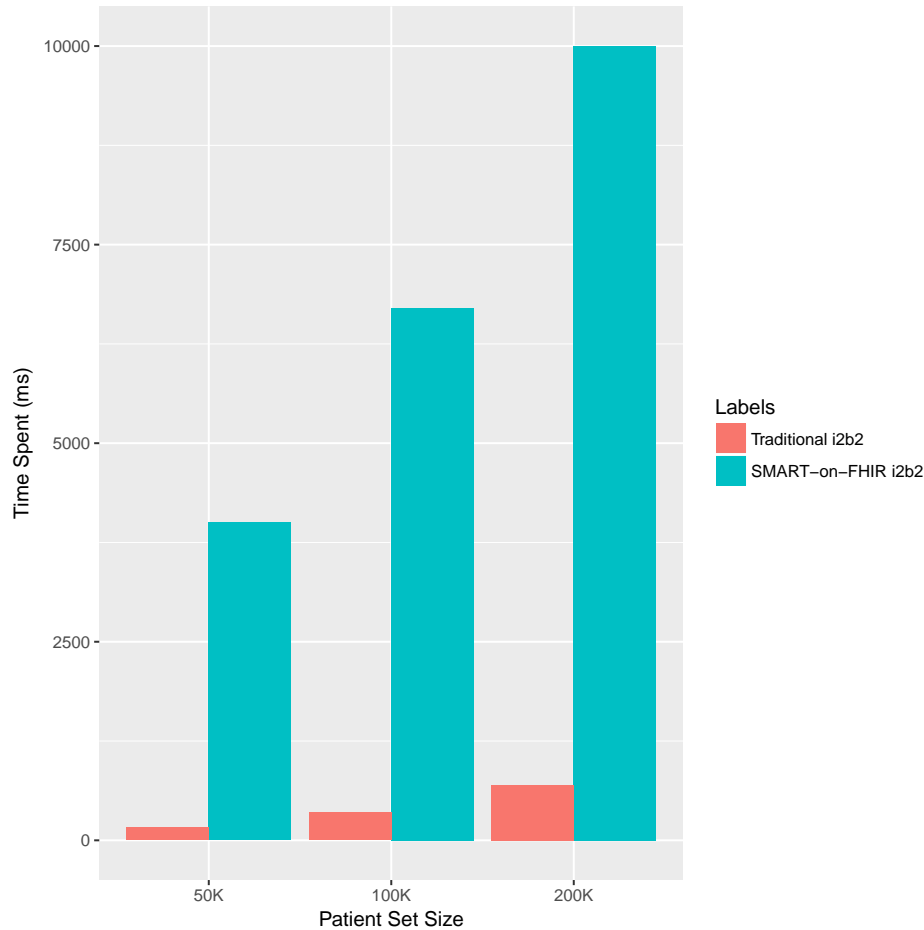
---
version: dstu3
Patient:
  patientUriPath: $.resource.id
  patientUriField: id
Observation:
  - patientUriPath: $.resource.subject.reference ,
  - encounterUriPath: $.resource.context.reference
  - instanceUriPath: $.resource.id
  - datePath: $.resource.effectiveDateTime
  - patientUriField: subject
  - encounterUriField: context
  - instanceUriField: id
  - dateField: effective
[...]
```

**Figure 3:** FHIR resource YAML configuration file

apache HIVE table on a big-data platform. The results show that the time spent are under the minute and compatible with i2b2 promizes. Moreover, the barplots show that the major bottleneck is the FHIR json Generation step. Such amount of data have never been described to be handle by i2b2 before since we approach here traditionnal RDBMS limitations volumetry. While traditionnal i2b2 outperforms the FHIR based one on modest datasets, the latter opens new perspectives by allowing to connect to specialized and optimized database systems.

*i2b2 feature coverage:* i2b2 querying feature covers filtering patients facts by code, values, dates, thought patient history, within an encounter temporal window or even a free sequence of events. By adding new temporal table mechanisms, the present work allows all those features. Ence, it does not limit the existing set of functionalities. The i2b2-FHIR configuration file Figure 3 contains information about the FHIR-API instance, such its version, and how the resources are implemented. Depending on the kind of cohort set, the user want to extract, patient ID, encounter ID, instance ID or dates are retrived from the FHIR-API thanks to a jsonPATH description. This then allows to populate the CRC temporary tables. This is how i2b2 deliberation mecanisms can be populated, and the set build. The Table ?? shows how those features are covered.

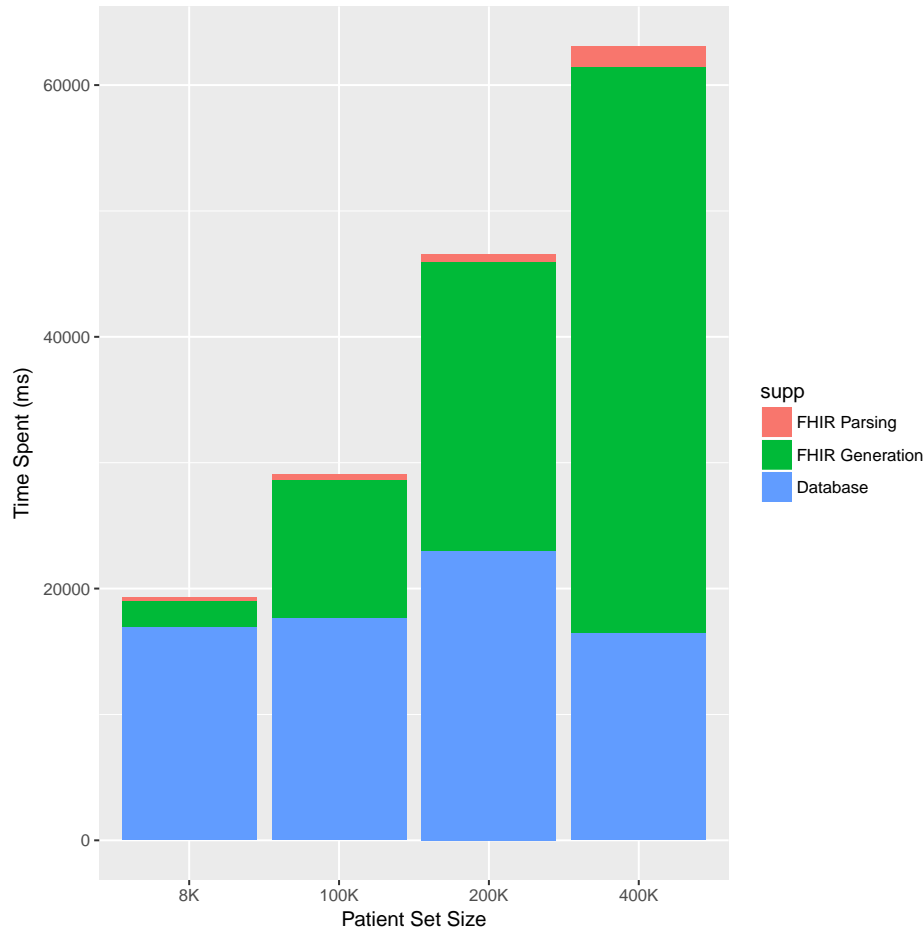
*Security:* A security layer has been proposed and implemented into the existing CRC cell. A new i2b2 table allows to define witch patient are part of wich project. This security layer is important because it allows with one endpoint with all patients records, to create multiple projects with subset. In terms of performances, the table might be vertivally partitionned and splitted by project, in order to get stable performances while number of project will increase. This mecanism is both compatible with traditionnal i2b2 and i2b2-SMART-on-FHIR and has been deployed in production and handle more than distinct 200 projects. The Oauth2 security layer has not yet been implemented. The implementation will inspire from project<sup>3,4</sup> that recently succeed in.



**Figure 4:** Traditionnal versus SMART-on-FHIR performances comparison (on a 150M postgresQL table)

*Extensibility:* The FHIR access layer has been tested over the HAPI FHIR test server for all resources at least to refereing to a patient (68 resources), and does have a complete resource coverage. To date, the query builder is compatible with last FHIR DSTU3 version. In the future, it will be compatible with each FHIR release, and maintain backward compatibilities. The FHIR version of each endpoint is setup in the configuration file (Figure 3). The query builder handles the FHIR extensibility, local profiled resources or even local new resources. Moreover, the design allows to filter based on FHIR extensions (<https://www.hl7.org/fhir/extensibility.html#extension>). The results let conclude the design is flexible enough to query multiple centers with different fhir implementation at the same time.

*Interoperability:* FHIR conceptMapping objects have been produced to make the proof of concept. The Table ?? shows the HTTP query used to retrieve the equivalent code. The conceptMapping are then parsed to only collect the "exact-matches" codes. The FHIR-search query is then extended with the collected code-systems/codes. The design allows to query multiple FHIR endpoints using different terminologies. However, this does not allow to match valuesets. While there is some field of improvement, the results open area for massive and collaborative concept mapping, with a terminology server ready. By querying over the conceptMapping resources, the CRC cell is able to expand the query to fetch all synonyms. This design allows implementers to align across different code systems, or to define mapping within the same code system. Since the FHIR-configuration allows to use a specific ConceptMapping endpoint, this allows to federate queries and use only one terminology server where concept mapping is done in collaboration.



**Figure 5:** SMART-on-FHIR performances (on a 5B Hive table)

## Conclusion

The main contribution of the work is to pave the way for cohort-generation process by leveraging standard access, with interoperable terminology systems and state of the art security methods. The hospital centers international effort to converge to FHIR data exchange layer[ref] will ease the data-federation to query center without dedicated datawarehousing staff.

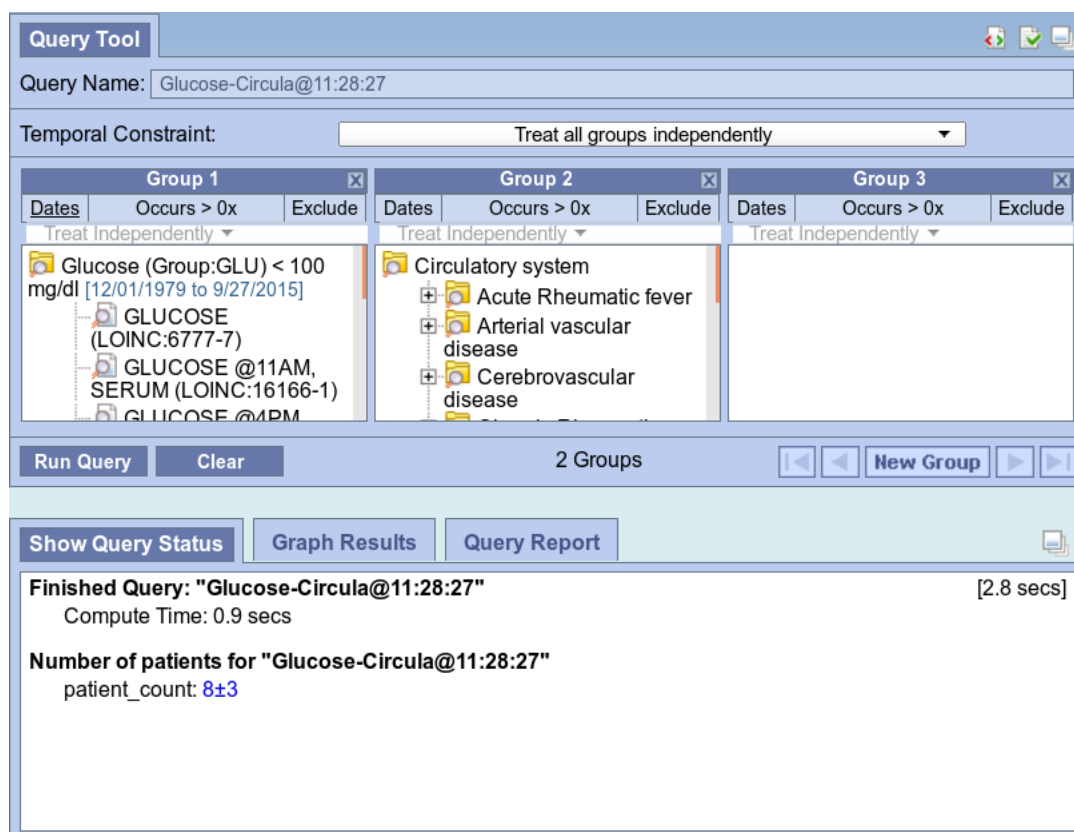
The secondary contribution of the work is to allow implementers to use their technology, and will allow i2b2 instance to benefit from the fast past and future improvements on big-data technologies.

All together, this work opens a new-area of specific solution to apprehend the medical diversity, variety and volumetry.

## Discussion

By introducing into i2b2 a dependency on FHIR specifications, this will lead to bring together community and improve both sides.

Several modules have been implemented, some aspects of the design have only been tested as separate modules. The roadmap provides for the development of multiple SMART-on-FHIR endpoints access, OAuth2 implementation and performances improvements. Once satisfied with the results, the system should be available in next releases of core i2b2. Specific exploration around specialized databases (temporal-series, text-mining, distributed, graph databases) will result to better handling variety of big-data, such as genomic[ref], textual notes, DICOM imaging, physiological



**Figure 6:** SMART-on-FHIR online demo screenshot

waveforms or exposomic.

While all resource containing patient reference were tested, there is a need to propose a general mapping between traditional i2b2 objects (patient, visit, provider, observation) and FHIR specific resources (Organization, Healthcare-Service, Patient, EpisodeOfCare, Condition, Procedure, Medication, MedicationRequest, Observation, DiagnosticReport, ClinicalImpression...). A general algorithm to translate FHIR terminologies into i2b2 ontology will also be investigated, and result as a complementary software.

Last but not least concept mapping between many institutions and languages remains to be done. Since all are based on different languages, different granularity and different concept and practices, this remains a challenge to be addressed. While ontology matching has a long exp, this research area is still challenging.

## References

1. Pryor TA, Gardner RM, Clayton RD, Warner HR. The HELP system. J Med Sys. 1983;7:87-101.
2. Gardner RM, Golubjatnikov OK, Laub RM, Jacobson JT, Evans RS. Computer-critiqued blood ordering using the HELP system. Comput Biomed Res 1990;23:514-28.
3. Waghlikar KB, Mandel JC, Klann JG, Wattanasin N, Mendis M, Chute CG, et al. SMART-on-FHIR implemented over i2b2. Journal of the American Medical Informatics Association. 2016 Jun 6;ocw079.
4. Pfiffner PB, Pinyol I, Natter MD, Mandl KD. C3-PRO: Connecting ResearchKit to the Health System Using i2b2 and FHIR. Seo J-S, editor. PLOS ONE. 2016 Mar 31;11(3):e0152722.
5. Alterovitz G, Warner J, Zhang P, Chen Y, Ullman-Cullere M, Kreda D, et al. SMART on FHIR Genomics: Facilitating standardized clinico-genomic apps. Journal of the American Medical Informatics Association. 2015 Jul 21;ocv045.