EHR-Driven Phenotyping

Improving Standards & Methods for Secondary Use of EHR Data

Pascal S. Brandt

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2021

Reading Committee:

Adam Wilcox, Chair

Bill Lober

John Gennari

Program Authorized to Offer Degree:
Department of Biomedical Informatics and Medical Education

University of Washington

Abstract

EHR-Driven Phenotyping

Pascal S. Brandt

Chair of the Supervisory Committee:
Adam Wilcox
Department of Biomedical Informatics and Medical Education

The digital transformation of healthcare over the past two decades has led to the proliferation of electronic health record (EHR) databases. These databases present an unprecedented opportunity for biomedical knowledge discovery. Data may be used for several purposes, including epidemiology, operational or clinical quality improvement studies, pragmatic trials and clinical trial recruitment, comparative effectiveness research, predictive modeling, clinical decision support, pharmacovigilance, and genome-wide association studies. In every case, one of the first steps involved is identifying the appropriate cohort of patients matching a set of inclusion and exclusion criteria, using only data available in the EHR. This process, known as *EHR-driven phenotyping*, is a resource-intensive task that involves many stakeholders, such as clinical experts, informaticists, and database analysts. It is therefore a critical rate-limiting factor that prevents massive scaling of knowledge discovery, and ultimately inhibits our ability to achieve the promise of national imperatives such as the *Learning Healthcare*

System and *All of Us*.

This research will attempt to improve the state of the art of EHR-driven phenotyping in three specific ways. First, we will analyze the variability of a set of existing, clinically validated, phenotype definitions in order to understand the requirements for a formal representation that supports automation. Second, we will assess the suitability of popular and emerging standards for formally representing cohort criteria, and evaluate whether this representation facilitates cross-platform cohort identification. Finally, we will develop and evaluate a fully standards-based system that can be used to create phenotype definitions and execute them against existing EHR data platforms, and evaluate the performance of this system in the context of the extant EHR-driven phenotyping ecosystem.

DEDICATION

This dissertation is dedicated to my mother, Linnette Brandt, for always believing in me.

ACKNOWLEDGEMENTS

I have been incredibly fortunate to have encountered so many amazing people during the course of this work. Each interaction and experience contributed something to me and therefore to this dissertation, and I acknowledge and thank everyone for their input.

Thank you to my committee chair, Adam Wilcox, for letting me follow my own interests, for opening doors by making the right connections, and for keeping me on track when I needed it. Thank you Bill Lober for the guidance you provided, for helping during the application process, and for the work we did together both before and during my time in the program. Thanks John Gennari for all of your detailed feedback and questions, and thank you Brian Shirts for serving as the graduate school representative on my committee.

I both learned so much from, and had a lot of fun with all of my colleagues in the BHI program. The five years we spent together will be a time I remember fondly for the rest of my life, and I look forward to continuing to learn from and create memories with many of you. I would especially like to thank Piotr Mankowski, Tim Bergquist, Lauren Snyder, Hannah Burkhardt, Aakash Sur, Maggie Dorr, and Chethan Jujjavarapu. My thanks also go to all the faculty and staff in the BIME department.

Without my collaborators, I would not have been able to complete any of the work presented here. A special thank you goes to Luke Rasmussen, who was always willing to

discuss ideas, review manuscripts and code, and who contributed enormously to every one of my dissertation aims. Similarly, thank you to the PhEMA AIM 1 team, who I learned a lot from, and who regularly gave me extremely valuable feedback, and also contributed to each research aim. Thank you Jennifer Pacheco, Prakash Adekkanattu, and Richard Kiefer.

To my other collaborators at Weill Cornell and Northwestern, thank you all for your feedback and contributions, and for everything you taught me. Thank you Fei Wang, Zhenxing Xu, Jie Xu, Yifan Peng, Natalie Benda, Sajjad Abedian, Evan Sholle, and Chengsheng Mao.

A special thank you also goes to the PhEMA PIs, Jyotishman Pathak, Guoqian Jiang, and Yuan Luo. Thank you for letting me join your project. It was an honor to work with each of you, and learn the many lessons you had to offer.

Thank you the United States and South African governments, who provided funding through the Fulbright Program and National Research Foundation respectively. Thank you also to Commure, for being patient and flexible while I completed this work.

Most importantly, I would like to thank my significant others. Both Jodi Allemeier and Allison Frey spent many years dealing with my lack of availability, distraction, and stress. Thank you for always being generous, kind, and supportive. I could not have got to this point without you. Thank you my brothers, Sheldon and Joshua, for your friendship and contributions to my life, all of which enabled me to complete this work. Finally, thank you to my mother, Linnette Brandt. Thank you for raising me well, despite me often not making that very easy to do. Thank you for your constant encouragement and unwaivering belief in my ability to achieve any goal I set for myself. This work was only possible because of everything you did. Thank you.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

ACT Accrual to Clinical Trials

AHRQ Agency for Healthcare Research and Quality

API application programming interface

AST Abstract Syntax Tree

BD2K Big Data to Knowledge

CD2H National Center for Data to Health

CDM common data model

CDS clinical decision support

CMS Centers for Medicare & Medicaid Services

CPT Current Procedural Terminology

CQL Clinical Quality Language

CSV comma separated values

eCQM electronic clinical quality measure

EHR electronic health record

ELM Expression Logical Model

eMERGE electronic Medical Records and Genomics

ETL extract transform load

FFI foreign function invocation

FHIR Fast Healthcare Interoperability Resources

HAPI HL7 Application Programming Interface

HCSRN Health Care Systems Research Network

HEDIS® The Healthcare Effectiveness Data and Information Set

HF heart failure

HITECH Health Information Technology for Economic and Clinical Health Act

HL7 Health Level Seven International

HQMF Health Quality Measure Format

ICD International Classification of Diseases

JSON JavaScript Object Notation

KAS Knowledge Artifact Specification

KNIME Konstanz Information Miner

LHS Learning Healthcare System

MAT Measure Authoring Tool

ML machine learning

NCATS National Center for Advancing Translational Sciences

NIH National Institutes of Health

NLP natural language processing

NM Northwestern Medicine

NYP NewYork-Presbyterian

OHDSI Observational Health Data Sciences and Informatics

OID object identifier

OMOP Observational Medical Outcomes Partnership

PCORnet National Patient-Centered Clinical Research Network

PhEMA Phenotype Execution and Modeling Architecture

QDM Quality Data Model

R The R programming language

RCT randomized controlled trial

SNOMED Systematized Nomenclature of Medicine

SQL Structured Query Language

SynPUF 1k Centers for Medicare & Medicaid Services' (CMS) Data Entrepreneurs' Synthetic Public Use File

TE thrombotic event

TTF Task-Technology Fit

VSAC Value Set Authority Center

WCM Weill Cornell Medicine

XML Extensible Markup Language

CHAPTER 1

INTRODUCTION

The idea of evidence-based medicine was popularized by Archie Cochrane in his 1972 book *Effectiveness and Efficiency*[1], and has become the modern best practice for delivery of healthcare. In the years since the book was published, thousands of randomized controlled trials (RCTs) have been published that collectively comprise the evidence used for medical decision-making. While evidence from RCTs is considered the gold standard, they are expensive and time-consuming to conduct, and in some cases fail to account for all relevant clinical complexities[2]. In 2007, the National Academy of Medicine (then called the Institute of Medicine) released a report summarizing a workshop held on evidence-based medicine[3]. The report identified many challenges facing healthcare research and delivery at the time, and proposed a new framework to deal with these challenges called the *Learning Healthcare System* (LHS). The LHS aims to improve the velocity of evidence generation and its translation into clinical practice. Recommendations provided to accomplish this include bridging the gap between clinical research and practice by using data collected during routine care for research, developing clinical decision support systems, and creating tools for data mining. In summary, the LHS aims to reduce the time involved, and increase the scale and efficacy of biomedical knowledge generation and its translation into improved healthcare practice.

Fortunately, the digital transformation of medicine, in part catalyzed by the HITECH Act and Meaningful Use incentives, has resulted in broad adoption of electronic health records

(EHRs). In many settings, adoption rates are approaching 100%[4]. The data assembled in these systems present an enormous opportunity for clinical research and the improvement of care. To this end, the National Institutes of Health (NIH) has established several initiatives aimed at capitalizing on this abundance of digital healthcare data. These initiatives include the National Center for Advancing Translational Sciences (NCATS)*, the National Center for Data to Health (CD2H)†, and the Big Data to Knowledge (BD2K) project[5], among others. In addition, former President Obama launched the *Precision Medicine Initiative* (now called *All of Us*[6]), which includes an objective to "build the evidence base needed to guide clinical practice."[7] These projects and innumerable other grant awards and research initiatives all contribute to the acceleration and scaling of biomedical knowledge generation.

Despite this explosion of research activity, it is estimated that it takes an average of 17 years (with significant variance) for healthcare research outputs to be translated into improved practice[9]. There are undoubtedly many steps in the process that could be optimized, but in this work we focus on methods for optimizing evidence generation, specifically in the case of studies using real-world data collected in the EHR. A universal first step in these types of studies (listed in table 1.1) is to identify a cohort of patients of interest. The sets of criteria that define these patient cohorts are referred to as *phenotype definitions* or (somewhat imprecisely) just *phenotypes*. The process of establishing these cohorts for a given research study is thus referred to as *EHR-driven phenotyping*. Several factors complicate this ostensibly simple task, including EHR data quality and completeness issues, inter and intra-site variability in clinical processes, data model incompatibilities, terminology or

*https://ncats.nih.gov/
†https://ctsa.ncats.nih.gov/cd2h/

Study Type	Use Cases
Cross-sectional	Epidemiological research
	Hospital administration/resource allocation
	Adherence to diagnostic/treatment guidelines
	Quality measurement
Association (case-control/cohort)	Genome-wide association studies
	Pharmacovigilance
	Identifying clinical risk factors and protective factors
	Clinical decision support
	Clinical effectiveness research
	Predictive modeling
Experimental	Clinical trial recruitment
	Pragmatic trials
	Adaptive/randomized, embedded, multifactorial, adaptive platform trials

Table 1.1: Applications of EHR-driven phenotyping across study types[8]

ontology differences between systems, and more[10,11].

While EHR-driven phenotyping is used in all the study types listed in table 1.1, there are some differences worth noting between these use cases. The first important difference is in the frequency that the phenotyping task is performed. At the low end, the EHR-driven phenotyping could be executed only once. For example, in genome-wide association studies and observational clinical effectiveness research, cohorts are only established once, and retrospective data is then used for analysis. For clinical trial recruitment, however, phenotyping might be done more regularly in order to identify patients as they become eligible. For clinical decision support, phenotyping might be done daily, or even in real time in order to detect newly developing or emergent conditions. The other important difference

between the use cases is the data source used. Less time-sensitive applications may use data extracted from live EHR systems and loaded into data warehouses, while real-time use cases may need to use the most up to date data available only in the EHR. Despite these differences, however, the actual criteria and phenotyping mechanism used for each use case are conceptually similar.

The challenges faced during EHR-driven phenotyping are exacerbated when phenotype definitions are shared between sites, which is a common occurrence, since research networks such as the electronic Medical Records and Genomics (eMERGE) Network[12–14], the National Patient-Centered Clinical Research Network (PCORnet)[15], and many others often aim to combine results from multiple sites in order to increase statistical power and cohort diversity. In these multi-site studies, phenotype definitions are usually developed at one site and distributed to other sites in the form of narrative descriptions and lists of codes from published terminologies. Sometimes flowcharts or pseudocode are also supplied, but directly executable computable artifacts are rarely included[16]. Phenotype definitions distributed in this way must therefore be manually translated into executable code at each site. This requires human interpretation of the narrative text, and may also require manual translation of code lists into local terminologies. These tasks are slow and error-prone, and are neither efficient nor scalable, as they must be repeated at each new site wishing to participate in a study.

To begin to address these inefficiencies, at least two approaches have been used, namely common data models (CDMs) and dedicated logic execution environments. Research networks including the Accrual to Clinical Trials (ACT) Network[17] and the Observational Health Data Sciences and Informatics (OHDSI) program[18] make use of the CDM approach.

This approach requires researchers to transform data into the specified CDM, and in some cases translate coded data. As a result of using a CDM, database queries can be shared across sites, but there is an implementation cost involved in translating the data into the CDM format, and queries cannot be shared between different CDMs. The dedicated logic execution environment approach requires phenotype definitions to be represented in a format that can be directly executed. This increases velocity and reduces potential for error, since no manual translation is required, but current methods do not make use of healthcare standards, and can sometimes be prohibitively challenging to implement[19]. Furthermore, these methods require significant data preprocessing in some cases.

To mitigate these and other issues, the Phenotype Execution and Modeling Architecture (PhEMA)[20] project was initiated. The PhEMA project arose due to a need, identified by the eMERGE Network, to develop EHR-driven phenotyping methods that are scalable and portable between systems. Work done by PhEMA includes defining desiderata for computable phenotype definitions[16], early assessment of phenotype definition complexity[21], assessing potential standards-based representations[22–27], and highlighting considerations for phenotype definition portability[19,28,29]. The overarching goal of this work is to further develop the work done by PhEMA, with a focus on standards and interoperability.

1.1 Dissertation Aims

This dissertation makes use of the Task-Technology Fit (TTF) model[30], and each of the three aims correspond to a component of this model, as shown in figure 1.1. We begin by determining the characteristics of the EHR-driven phenotyping task. We then evaluate

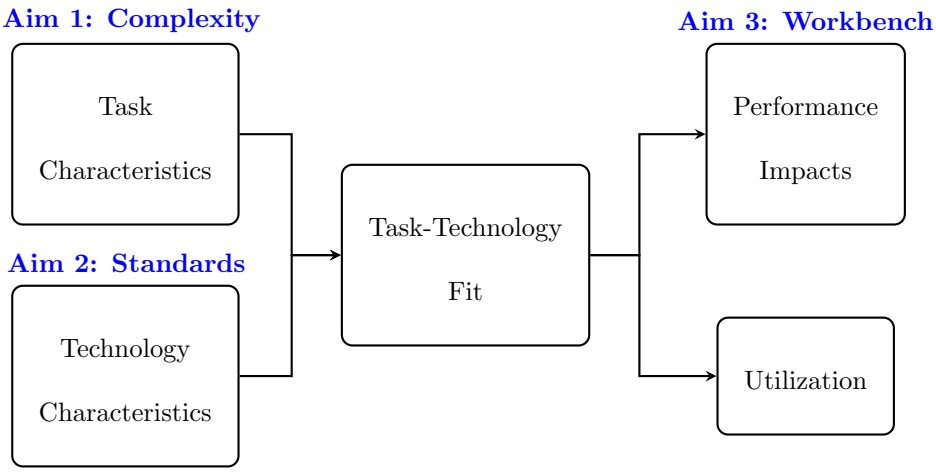


Figure 1.1: Aims in relation to Task-Technology Fit (TTF) model[30]

the feasibility of using popular and emerging technologies for representing and executing phenotype logical criteria, and finally we develop and evaluate the performance of a fully standards-based phenotype representation and associated open-source tool.

Aim 1: Characterizing the Dimensions of Phenotype Definition Variability

In the first aim we investigate the nature of EHR-driven phenotype definitions by examining a data set of phenotype definitions extracted from a repository of clinically validated definitions that have been used in published biomedical research. We describe the components that comprise a phenotype definition and analyze the dimensions along which the phenotypes in the data set vary. We also identify some important requirements that must be satisfied by any potential formal representation.

Aim 2: Toward Cross-Platform Electronic Health Record-Driven Phenotyping Using Clinical Quality Language

In the second aim we assess whether the Fast Healthcare Interoperability Resources (FHIR) and Clinical Quality Language (CQL) standards can be used to enable scalable and portable

phenotype definitions. We use CQL to represent phenotype logical criteria and execute these criteria using a FHIR server and a custom developed open-source tool that enables the execution of standards-based phenotype definitions against an Observational Medical Outcomes Partnership (OMOP) database[18]. We use this tool as part of an experiment spanning two academic medical centers and report the results.

Aim 3: PhEMA Workbench: A Platform-Independent FHIR-Native EHR-Driven Phenotyping Toolbox

In the final aim we propose a fully FHIR-native phenotype representation and develop and evaluate an interoperable EHR-driven phenotyping system. We integrate directly with the phenotype repository examined in the first aim, and incorporate the tool developed in the second aim. We demonstrate additional tools that facilitate phenotype authoring, including both logic and terminology representation, and demonstrate how this suite of tools interoperates with existing systems in the phenotyping and informatics ecosystem. We highlight how this approach achieves acceptable results while drastically reducing implementation time and potential for human error.

1.2 Dissertation Overview

Each of the three dissertation aims are investigated individually, and a standalone research paper is presented here for each. The papers for aims 1 through 3 are given in chapters 2 through 4 respectively, and are followed by a concluding chapter (chapter 5) in which we synthesize the contributions and discuss the limitations of this work, as well as identify potential avenues for further investigation. We believe that the methods we have developed

and evaluated in this work provide an incremental but significant contribution towards the goal of achieving truly scalable EHR-driven phenotyping.

CHAPTER 2

CHARACTERIZING THE DIMENSIONS OF PHENOTYPE DEFINITION VARIABILITY

2.1 Introduction

Many different types of research studies are used to generate biomedical knowledge from electronic health record (EHR) data[8], all of which first require establishing a cohort of patients meeting specific criteria. This cohort identification process is referred to as *EHR-driven phenotyping*, and the sets of inclusion and exclusion criteria are known as *phenotype definitions*, or just *phenotypes* (the term used in this manuscript for brevity). While often developed and executed at a single institution, research networks such as the National Patient-Centered Clinical Research Network (PCORnet)[31], the electronic Medical Records and Genomics (eMERGE) Network[12–14], and the Observational Health Data Sciences and Informatics (OHDSI) program[18] have run studies in a distributed manner to pool their results to improve statistical power and cohort diversity. To facilitate this type of federated study, phenotypes must be shared with and implemented at all participating sites.

Historically, phenotypes have been shared between sites via repositories like the Phenotype KnowledgeBase (PheKB)[32] in the form of narrative descriptions, sometimes accompanied by flowcharts or pseudocode. Lists of codes from common terminologies like the International Classification of Diseases version 9 (ICD-9) are usually also included, but in

most cases, directly computable artifacts, such as SQL scripts or programming code, are not. This method of phenotype distribution has proven to be a major limiting factor in the scaling up of biomedical knowledge generation[33–35], since implementing sites must manually interpret narrative descriptions to produce queries that can extract patient cohorts from local data sources. This process is time-consuming and error-prone, which is compounded by the fact that narrative descriptions can be ambiguous or difficult to interpret. To address these issues, the Phenotype Execution and Modeling Architecture (PhEMA)* project was established to optimize the EHR-driven phenotyping task, and has recommended that phenotype definitions be represented in a computable format[16]. Such a format would eliminate ambiguity and potentially facilitate automated cohort identification.

However, there is currently no widely accepted standard for representing computable phenotype definitions, although several have been proposed and evaluated[22–27,36,37]. In this study, we analyze a data set of phenotype definitions from PheKB and characterize their variation along several dimensions. Our goal is to provide an informational resource for EHR-driven phenotyping practitioners and researchers that will highlight important considerations for implementation, as well as contribute to identifying the requirements for a potential formal representation standard.

2.2 Background

To address the lack of scalability associated with implementing phenotype definitions shared as narrative descriptions, it has long been a goal of the EHR-driven phenotyping community to represent phenotypes in a computable format. Many different formats have been studied,

*https://projectphema.org

for example, the Health Quality Measure Format (HQMF) and Quality Data Model (QDM) have been used in conjunction with the JBoss® Drools engine[22,23] and the KNIME workflow execution engine[24,25]. In the United Kingdom, the Common Workflow Language (CWL) has been used to model and execute phenotype definitions[26]. Additionally, PhEMA researchers have demonstrated the feasibility of using Fast Healthcare Interoperability Resources (FHIR) and the Clinical Quality Language (CQL) for EHR-driven phenotyping[27,36–38]. While all of these studies are valuable and informative, they do not examine the nature of a broad range of phenotype definitions in detail.

That said, previous work has been done on the nature of eligibility criteria for clinical trials, which is one application of EHR-driven phenotyping[39,40]. These studies describe the high-level categorizations of the elements that make up clinical trial inclusion and exclusion criteria, such as the instances and combinations of Boolean and temporal operators, as well as the data elements used. The authors note that criteria provided as narrative text can sometimes be "incomprehensible", which results in errors and inefficiencies at implementation time, and call for "clear standards." A comprehensive review compares and contrasts 27 clinical trial criteria knowledge representation tools and models in considerable detail, and describes the range of criteria that can be represented using each[41]. However, this study focuses on tools, rather than real-world criteria, and none of the above studies focus on the more general task of EHR-driven phenotyping, which may have different requirements[42]. Criterion complexity has also been studied in the related area of clinical quality measurement[43,44], and it was noted that "some modifications" to the QDM are required to represent robust phenotype definitions.

A 2011 study analyzed the heterogeneity and complexity of 14 phenotype definitions

produced by the eMERGE network[21]. A significant amount of homogeneity was found among the set of phenotypes examined, which, according to the authors, suggests that a computable representation is feasible. The study focused mainly on text analysis of narrative descriptions and Boolean and temporal logic, but also described data types and terminologies used. A more recent study analyzed the effort required to implement a set of 55 phenotype definitions from the eMERGE network, and proposed a scoring system based on **k**nowledge conversion, logical clause **i**nterpretation, and **p**rogramming (KIP)[45]. This study focused on portability of phenotype definitions, and highlights many of the challenges currently faced by implementation sites. The study notes that it can take hours to months to implement phenotype definitions and that logic can become complicated when clauses are combined, and provides a detailed enumeration of the tasks required during implementation.

Despite the body of research described above, the EHR-driven phenotyping community has not yet settled on a single formal representation. This hinders not only the process of authoring and execution, but also the broader evaluation of how phenotyping (as a process) and the phenotypes themselves have changed over time. Only the 2011 study describes the variability of EHR-driven phenotype definitions, and only in relatively coarse detail using narrative descriptions. Many more phenotype definitions have been created in the decade since that study was conducted, and we expect that the complexity of these definitions has increased. Therefore, in this study we use a single formal representation to author multiple phenotype definitions, and leverage the benefits of this single representation to analyze a larger set of phenotype definitions in detail. This will allow us to provide insights into the representativeness of a formal definition for phenotyping, as well as the variability of the phenotypes themselves.

2.3 Materials & Methods

2.3.1 Data Set

We chose to analyze phenotypes from the PheKB phenotype repository as it is the most mature and widely used in the United States. PheKB was initiated in 2012 and has been continuously contributed to by various research teams, most notably by researchers involved in the eMERGE Network. The repository contains over one hundred phenotypes in various stages of development.

2.3.2 Data Extraction

A web scraping tool was developed to download all public phenotypes and associated files, including PDFs, Microsoft Word and Excel documents, images, ZIP files, and any other artifacts associated with the phenotype definition. Each phenotype in PheKB has a dedicated page that contains metadata curated by the phenotype authors, including the authors' names and research network affiliation, the demographics to which the phenotype applies, and more. Each phenotype optionally also includes one or more implementation reports, which provide a summary of the results for a specific implementation of the phenotype definition at a single institution. All metadata was stored in a JSON document alongside the downloaded artifacts to enable computational analysis. Source code for this step of the process is available on GitHub *.

*https://github.com/PheMA/phekb-export

2.3.3 Data Preparation

2.3.3.1 Phenotype Selection

From the full collection of phenotypes in PheKB, we used available metadata to automatically include those that were publicly available and marked with a status of "FINAL". We reviewed these phenotype definitions (descriptions, artifacts, and metadata) and only included those that used some structured data element (i.e., were not entirely natural language processing (NLP)-based), were an actual phenotype definition (e.g., did not simply serve as a repository to submit data), and were used in a published research study. These criteria were chosen to ensure that our analysis was conducted using only completed and clinically validated phenotype definitions.

2.3.3.2 Translation

In order to eliminate ambiguity and have a consistent, semantically correct, and computationally comparable representation of each selected phenotype definition, we chose to represent all phenotype definitions using FHIR and CQL. We selected FHIR because it is widely used and has recently become the legally required standard for clinical data exchange in the United States. Additionally, the Common Data Model Harmonization (CDMH) project[46] provides mappings from FHIR to many other common healthcare data models, maximizing the potential impact of the set of translated phenotype definitions. The entities referenced in each phenotype were modeled as FHIR resources such as `Patient` for individuals, `Observation` for labs and vitals, `Condition` for diagnoses, and `MedicationRequest` for drug orders.

Standard terminologies such as the International Classification of Diseases versions 9 (ICD-9) and 10 (ICD-10), Current Procedural Terminology (CPT), Logical Observation

Identifiers Names and Codes (LOINC®), and RxNorm were used for coded data, and lists of codes were represented using FHIR `ValueSet` resources. These terminologies were usually explicitly specified in the phenotype definitions, but where they were not, we used the recommended default terminologies from the FHIR standard. We developed an open-source tool to translate value sets in various formats into FHIR resources, and built an interface to allow web-based interaction with the tool*. The tool can translate CSV files, as well as concept sets exported from the OHDSI platform, into `ValueSet` resources. It also supports searching, inspecting, and importing value sets directly from the Value Set Authority Center (VSAC)[47], using the VSAC FHIR server.

Phenotype definition logic was represented using CQL, which has been shown to be a feasible logical expression language for representing clinically validated phenotypes[36,37,48–51]. CQL supports a wide range of Boolean, temporal, aggregate, and other operations. The language is data model independent, but works out of the box with FHIR. For each phenotype we created a single CQL library that contained the logic required to identify a matching patient. Logic shared between phenotypes was authored in shared libraries that were imported using the CQL `include` operator.

We did not implement NLP logic, as there is no widely accepted standard representation or implementation of this type of logic. To our knowledge, there is currently no way to natively express NLP constructs using FHIR or CQL, although this is an active area of research[27,38]. However, we did annotate which phenotypes make use of NLP. We additionally chose not to include an analysis of phenotype description narrative text in our evaluation. While previous authors have done this[21], we found that the narratives uploaded to PheKB

*https://github.com/PheMA/terminology-manager

vary widely, with some authors providing only a few sentences and others uploading the complete journal article. Importantly, we found that these descriptions did not correlate with other phenotype attributes such as number of logical expressions or value sets, so we felt that including analysis dimensions such as word count and sentence complexity would not contribute meaningful information to our results.

We adopted a number of conventions for the standards-based representation. First, in this work we only represent phenotype cases, and not controls, suspected cases or subtypes. Case definitions usually contain the most and most varied criteria, so serve as a good basis for comparison. We adopted the convention of creating a CQL statement in each library called "Case", which represents the entry point for evaluating the phenotype definition. Additionally, unless explicitly stated otherwise, we modeled drugs using their RxNorm ingredient name and lab values using the highest ranked appropriate LOINC® code. Although there is a possibility that these modeling choices may be semantically incorrect or suboptimal, this does not affect the primary objective of this work, which is phenotype definition comparison.

Following these conventions, two authors (PB and LR) independently translated each of the phenotypes using the available metadata and artifacts downloaded from PheKB. One author was primarily responsible for the translation of each phenotype, but the authors were not entirely blinded. Consultation amongst the larger study team was needed to confirm interpretation of phenotype definitions that were ambiguous.

2.3.3.3 *Development Environment*

We made use of several open-source tools during the phenotype translation process and published our tool chain configuration to the project GitHub repository*. We used Visual Studio Code† as our primary development environment, and for CQL syntax highlighting we used the `language-cql` plugin‡.

To translate CQL into the equivalent machine-readable representation, known as the Expression Logical Model (ELM), we used the reference implementation of the CQL to ELM translator§. For testing, we used the CQL Testing Framework (CTF) developed by the Agency for Healthcare Research and Quality (AHRQ)¶, which provides a mechanism to specify test data, which are materialized as FHIR resources, using a simple YAML file. The CTF also provides a configurable test runner, which can run a specific CQL library against the test data generated by the YAML specification, and assert that the results match what is expected.

2.3.4 Validation

We used two methods to ensure that phenotype definitions represented in FHIR and CQL were correctly translated from the artifacts available in PheKB. First, each phenotype was translated by a single author, and then verified using a code review process. The initial author created a pull request on GitHub (a way of isolating code for a specific purpose, in this case representing a single phenotype definition), and a second author reviewed the code to make sure it accurately represented the phenotype definition as described in PheKB.

*https://github.com/PheMA/phekb-phenotypes
†https://code.visualstudio.com
‡https://github.com/Jonnokc/Clinical-Quality-Language
§https://github.com/cqframework/clinical_quality_language
¶https://github.com/AHRQ-CDS/CQL-Testing-Framework

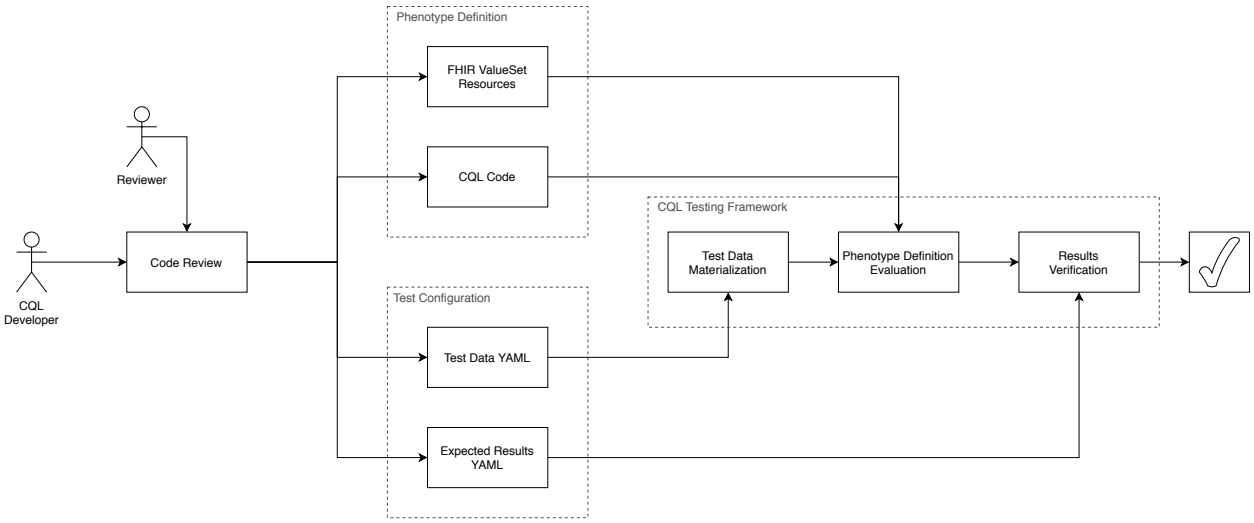


Figure 2.1: Development and validation pipeline.

Secondly, we used an approach from software engineering called test-driven development (TDD) to ensure that our translations of phenotype logic and value sets were correct. We made use of the CTF to implement this approach. In addition to allowing the CQL author to express both test cases and FHIR data using YAML, the CTF integrates with the Mocha JavaScript testing framework* in order to evaluate phenotype logic using the given data, and to assert that results produced are correct. This evaluation is done using the ELM representation of the phenotype, and the open-source JavaScript CQL engine †.

All tests were run automatically on each code commit to ensure no regressions were introduced. The full development and validation pipeline is shown in figure 2.1.

*https://mochajs.org
†https://github.com/cqframework/cql-execution

2.3.5 Data Analysis

2.3.5.1 Metadata Analysis

Two authors (PB and LR) independently conducted a manual review of both the published artifacts and metadata for each selected phenotype in order to identify relevant dimensions, emergent patterns, and characteristics not explicitly captured by PheKB. We categorized the artifacts provided with each phenotype definition (e.g., flowchart) and whether or not the definition for controls, subtypes, or suspected cases is provided. We also provide a brief description if the phenotype is underspecified (lacks enough detail to implement) or requires local knowledge (e.g., how "follow up" is defined). We also provide our own "Type" categorization created to capture the intent of the phenotype. We capture whether or not the phenotype uses tabular data, and how this data is provided. In most cases tabular data refers to lists of codes from standard terminologies, but also includes lists of keywords or medication names. Following this manual review, the authors met to discuss their findings and resolved any discordant determinations. We additionally conducted a computational analysis of the metadata JSON files extracted using the web scraping tool described above. This analysis was done using the Python programming language and Jupyter Lab notebooks.

2.3.5.2 Phenotype Definition Analysis

In order to evaluate the phenotype definition logic, we conducted an automated analysis of the ELM representation of each CQL library. The ELM is an instance of what is known in computer science, more specifically in programming language development, as an Abstract Syntax Tree (AST)[52]. The intention of an AST is to act as a machine-readable representation of a complete program, and is often used to evaluate or execute the program. However,

Category	Description	Examples
Aggregate	Operations that calculate single values from collections	`Sum()`, `Count()` or `Mean()`
Arithmetic	Mathematical operations	$+$, $-$ or $*$
Collection	Operations on collections of data like sets and lists	`First()`, `exists()` or `union`
Comparison	Numeric or date comparisons	$>$ or $=$
Conditional	Branching logic	`if` or `case`
Data	Data retrieval and filtering operations	FHIR resource retrieval and filtering by value set
Expressions	Total number of expressions as well as their depth	Total expression count, `where` clause expression depth
Literals	Explicit values, codes and quantities	23, `5 months` or `0.5 mg/dL`
Logical	Boolean logical operators	`and` or `not`
Temporal	Operators relating to dates and times	`before`, `starts` or `overlaps`
Terminology	Number of value sets used and the number of individual codes	Value sets per phenotype, codes per code system

Table 2.1: Phenotype definition analysis dimensions.

ASTs can also be used in program translation, as has been shown for CQL[37], or for program analysis, as we demonstrate here. We evaluate the ELM for each translated phenotype by making use of the Visitor Pattern[53], which is a mechanism for inspecting each node of tree-like data structure, and executing custom code in the context of each node. We implemented the Visitor Pattern in the Java programming language by using an interface provided by the reference implementation of the CQL translator and published it to GitHub*. The interface used is the same as the one used by the CQL engine during program execution. Using this implementation, we are able to calculate a number of measures about a given CQL library, such as how many value sets are referenced, how many Boolean, temporal, and aggregate operators are used, as well as how these operators are combined. It is also possible to count the total number of expressions, how many data types are used, and how many unique data queries are performed. After a manual review of the phenotype definitions, we identified 11

*https://github.com/PheMA/elm-utils

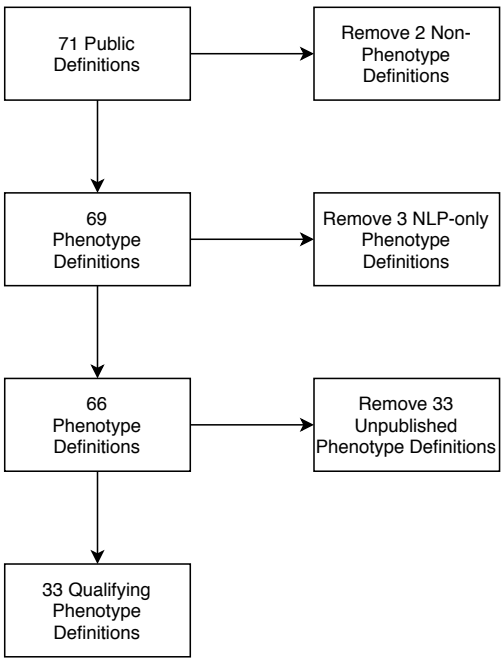


Figure 2.2: Phenotype definition selection process.

dimensions along which to evaluate each phenotype definition, shown in table 2.1.

2.4 Results

2.4.1 Phenotype Selection

At the time of our analysis there were a total of 71 publicly available phenotype definitions in PheKB with a status of "FINAL". We excluded 2 definitions that were not actually phenotypes. One was used as a placeholder to publish new value sets, and one was the description of a risk model. We eliminated three more that used only NLP criteria. Finally, from the remaining phenotypes we included only those with associated publications. This selection process, which resulted in 33 total phenotype definitions, is illustrated in figure 2.2.

2.4.2 Phenotype Artifacts

During the translation process, we created 40 CQL libraries - one for each phenotype, and 7 helper libraries, totaling 3,327 lines of CQL code. A total of 231 value sets were assembled, of which 216 were manually created and 15 were imported from VSAC. These value sets consist of 17,948 individual codes, of which 13,340 are unique. Additionally, 347 test cases were written that collectively contain 2,044 test assertions. To support these test cases, 347 patients, 96 encounters, 101 procedures, 335 medication orders, 385 conditions, and 360 observations were manually created as FHIR resources using the CTF.

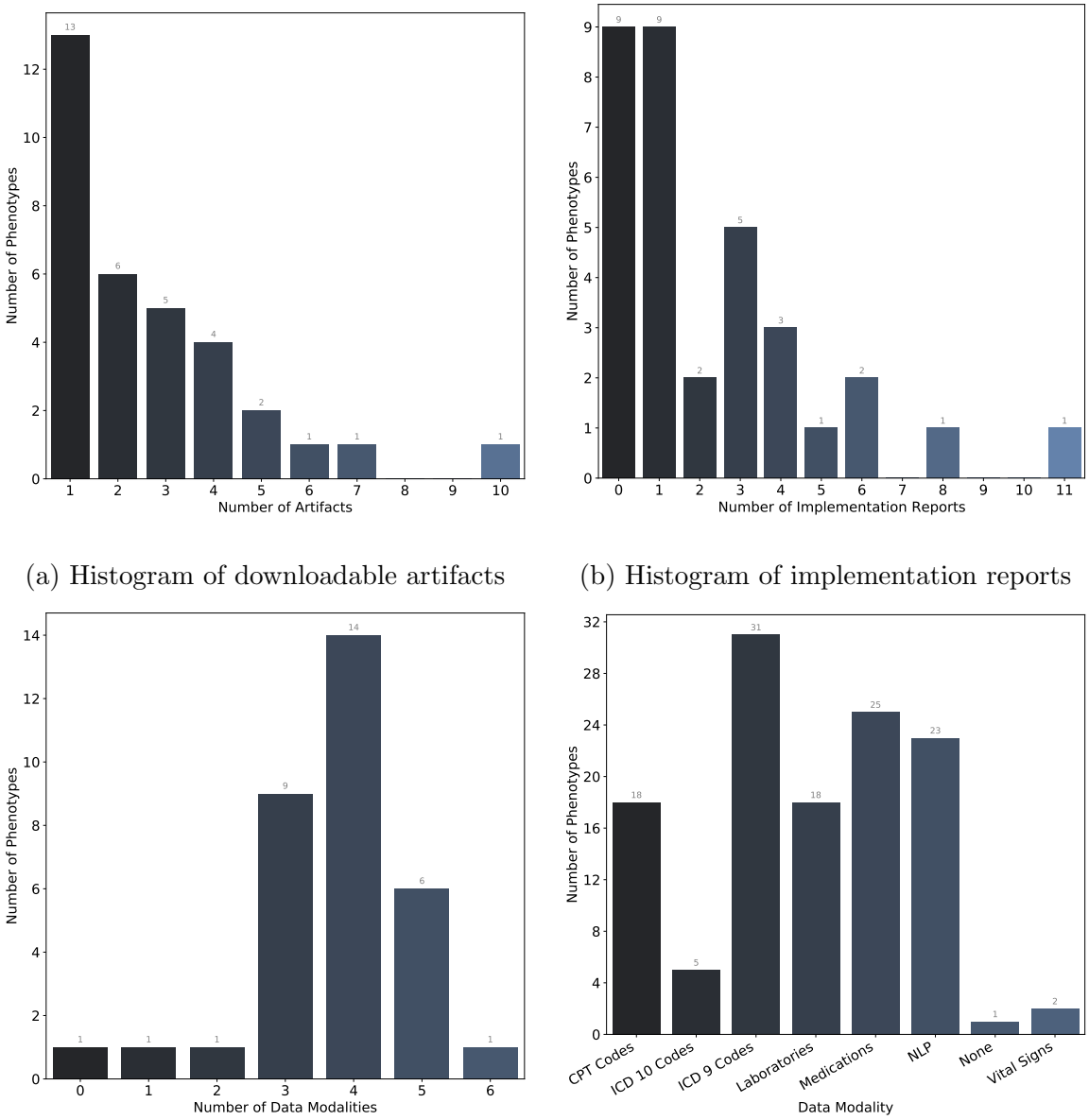
2.4.3 Metadata

Table 2.2 provides all self-reported metadata from PheKB. Histograms of some of the more complete metadata attributes are given in figure 2.3.

Name	Organizations	Networks	Authors	Date	Types	Artifacts	Results	Data	Gender	Ethnicity	Races
Asthma Response to Inhaled Steroids [54]		PGPop	1	2012-06-25	DR	2	0				0
Atrial Fibrillation [55]	VU		2	2012-03-20	DS	1	1	C; I9; NLP			0
Autism [56]	CCH	eMERGE	1	2013-04-16	DS	4	2	I9; M; NLP	F; M		0
Benign Prostatic Hyperplasia [19]	NU	eMERGE	2	2018-07-20	DS	1	0	C; I9; L; M	M		0
Bone Scan Utilization [57]	SU	NIH Collaboratory	7	2019-04-25	OT	1	0	C; I10; I9; NLP	M		0
Cardiac Conduction [58–61]	VU	eMERGE	1	2012-02-06	OT	4	4	C; I9; L; M; NLP	F; M		0
Cataracts [62–64]	MCRF	eMERGE	3	2012-02-06	DS	3	3	C; I9; M; NLP			0
Clopidogrel Poor Metabolizers [65]	VU		3	2012-04-03	DR	2	1	C; I9; L; M; NLP	F; M		0
Crohn's Disease [55]	VU		2	2012-03-20	DS	1	1	I9; M; NLP			0
Developmental Language Disorder [66]	VU		0	2020-07-27	DS; OT	6	0	I10; I9	F; M	H; NH	10
Digital Rectal Exam [67,68]	SU		6	2019-05-13	OT	1	0	C; I10; I9; NLP	M		0
Drug Induced Liver Injury [69,70]	CU	eMERGE	2	2012-12-07	DR	2	6	I9; L; M; NLP			0
Familial Hypercholesterolemia [71]	MC	eMERGE	10	2016-11-10	DS	7	11	C; I9; L; M; NLP	F; M	H; NH	0
Height [72]	NU		2	2012-02-06	OT	3	3	I9; L; M			0
Herpes Zoster [73]	GH; UW		1	2012-06-24	DS	5	5	C; I9; M; V	F; M	H; NH	10
High-Density Lipoproteins [74,75]	MCRF	eMERGE	2	2012-02-06	OT	1	1	I9; L; M; NLP			0
Hypothyroidism [76,77]	GH; MCRF; MC; NU; VU	eMERGE	1	2012-02-06	DS	3	6	C; I9; L; M; NLP			0
Lipids [78]	NU	eMERGE	2	2012-02-06	OT	3	3	I9; L; M			0
Multimodal Analgesia [79]	SU		2	2017-07-01	OT	1	0	C; I10; I9; M			0
Multiple Sclerosis [55]	VU		2	2012-03-20	DS	2	1	I9; M; NLP			0
Peripheral Arterial Disease [80]	MC	eMERGE	1	2012-02-06	DS	4	3	C; I9; L; M; NLP			0
Red Blood Cell Indices [81]	MC	eMERGE	1	2012-02-06	OT	3	4	C; I9; L; M; NLP			0
Resistant hypertension [82]	VU	eMERGE	2	2012-03-12	DR	4	1	C; I9; L; M; NLP; V	F; M		0
Rheumatoid Arthritis [55]	VU		2	2012-03-20	DS	1	1	I9; M; NLP			0
Sickle Cell Disease [83]	MCW	PCORI	2	2017-01-03	DS	1	2	I9	F; M		0
Statins and MACE [84,85]	VU	eMERGE; PGPop; PGRN	1	2013-06-07	DR	5	8	C; I9; L; NLP	F; M		0
Steroid Induced Osteonecrosis [86]	VUMC	PGRN	1	2013-03-25	DR	1	0	C; I9; M; NLP	F		0
Systemic Lupus [87]	VU	PCORI	6	2016-07-07	DS	1	0	I9; L; M; NLP	F; M	H; NH	8
Type 2 Diabetes [55]	VU		2	2012-03-20	DS	2	1	I9; L; M; NLP			0
Type 2 Diabetes Mellitus [88–90]	NU	eMERGE	2	2012-02-06	DS	10	4	I9; L; M			0
Urinary Incontinence [91]	SU		8	2020-01-15	DS	1	0	C; I10; I9; NLP	M	H; NH	10
Warfarin Dose/Response [92]	VU		2	2013-03-25	DR	1	1	L; M; NLP			0
White Blood Cell Indices [93]	GH	eMERGE	2	2012-02-06	OT	2	3	C; I9; L; M			0

Table 2.2: Self-reported metadata from PheKB.

VU – Vanderbilt University Medical Center, **MCRF** – Marshfield Clinic Research Foundation, **NU** – Northwestern University, **GH** – Group Health, **UW** – University of Washington, **SU** – Stanford University School of Medicine, **MC** – Mayo Clinic, **CU** – Columbia University, **MCW** – Medical College of Wisconsin, **CCH** – Cincinnati Children's Hospital Medical Center, **OT** – Other Trait, **DS** – Disease or Syndrome, **DR** – Drug Response - adverse effect or efficacy, **C** – CPT Codes, **I9** – ICD 9 Codes, **I10** – ICD 10 Codes, **M** – Medications, **L** – Laboratories, **V** – Vital Signs, **NLP** – Natural Language Processing, **F** – Female, **M** – Male, **H** – Hispanic, **NH** – Non-Hispanic, **PGPop** – Pharmacogenomic Discovery and Replication in Very Large Patient Populations, **eMERGE** – The Electronic Medical Records and Genomics Network, **PCORI** – Patient-Centered Outcomes Research Institute, **PGRN** – Pharmacogenomics Research Network



(a) Histogram of downloadable artifacts

(b) Histogram of implementation reports

(c) Histogram of data modalities

(d) Data modality utilization

Figure 2.3: Histograms of a selection of self-reported metadata.

Of 51 unique authors, most (36) only contributed to a single phenotype definition, while one author contributed to 10, which is twice as many as any other author. About half of the definitions (15) were associated with the authors from the eMERGE Network and the majority (20) were added in 2012. Most phenotypes (24) have 3 or fewer artifacts and about half (18) provide one or zero implementation reports. The vast majority (26) report using 4 or fewer data modalities, with only one (*Resistant Hypertension*) using all six modalities. ICD-9 codes were the most common data modality (31), followed by medications (25), and NLP (23). The *Asthma Response to Inhaled Steroids* phenotype reported using zero data modalities, but we found that it uses conditions, medications, and encounters. Data for race, gender and ethnicity were reported in under half of the phenotypes, with only 4 phenotype definitions mentioning race.

Table 2.3 provides additional metadata extracted by manually reviewing each phenotype definition. Our manually determined types mostly align with the PheKB types, but we introduce a new type with the label "Healthy / Valid Data". This indicates that the phenotype intends to identify healthy patients with valid data. For example, the *Height* phenotype identifies patients that have a valid height measurement and do not have any conditions that may impact height. We also introduce the "Treatment / Therapy" type, which identifies patients that have had a specific treatment, for example, *Bone Scan Utilization*. Finally, we report the approach described for implementing NLP (if applicable).

About two thirds of the definitions provide a narrative description (20) and flowchart (19), while only about one third (12) provide pseudocode. While tabular data is provided by all but three phenotypes, only 4 provide this data in a computable format. Computable artifacts in the form of KNIME workflows are provided for 5 phenotypes, and we found

that these workflows require users to prepare their data in a specified custom format before execution.

Most phenotypes (20) provide control definitions, 8 provide subtype definitions, and 4 define suspected cases. About half (16) of the phenotypes provide a list of covariates to be collected. Most (21) phenotypes are either underspecified or require some form of site-specific knowledge to fully implement. All but 5 phenotypes rely on some form of NLP, with 17 providing a list of keywords, 8 providing regular expressions, and 6 providing a list of medication names.

Name	Type	Narrative	Flowchart	Pseudocode	Tabular	Executable	Suspected Cases	Controls	Subtypes	Local/Underspecifed	Covariates	NLP
Asthma Response to Inhaled Steroids	Drug Response	✓			NC					“steroids are limited to WIZ orders”		regex
Atrial Fibrillation	Disease			✓	NC		✓	✓	✓			keywords; regex
Autism	Disease	✓	✓		NC			✓	✓	Does not specify what to do with DSM-IV criteria	✓	DSM-IV criteria
Benign Prostatic Hyperplasia	Disease		✓	✓	NC	KNIME		✓			✓	keywords
Bone Scan Utilization	Treatment / Therapy	✓	✓		NC			✓		“surgical procedure [...] as identified by [...] clinical notes”		keywords
Cardiac Conduction	Trait	✓			NC					“most recent clinic visit”	✓	keywords; negation; uncertainty
Cataracts	Disease		✓	✓	NC			✓	✓	“subjects where questionnaires have been scanned”	✓	MedLEE concepts; negation; regex
Clopidogrel Poor Metabolizers	Drug Response		✓		NC			✓	✓	Underspecified definition of “follow up”		keywords
Crohn’s Disease	Disease			✓	NC		✓	✓	✓			keywords; regex
Developmental Language Disorder	Disease	✓	✓		CSV	KNIME			✓	“pediatric records” not clearly defined		
Digital Rectal Exam	Treatment / Therapy	✓	✓		NC			✓		“physician documented notes”		”documentation is obtained from clinical notes”
Drug Induced Liver Injury	Drup Response	✓	✓		NC			✓		“Consider chronicity” not defined	✓	medication names; ”diagnosis mentioned”
Familial Hypercholesterolemia	Disease	✓	✓	✓	XLSX			✓			✓	detailed pseudocode
Height	Healthy / Valid Data		✓		NC	KNIME					✓	keywords
Herpes Zoster	Disease		✓	✓	XLSX			✓		“Implementations [...] may vary by institution”	✓	
High-Density Lipoproteins	Trait		✓	✓	NC					“cancer diagnosis in registry”		keywords
Hypothyroidism	Disease	✓			NC			✓		“requirement for annual physical”	✓	keywords
Lipids	Healthy / Valid Data	✓	✓	✓	NC	KNIME				“Genotyped pts”	✓	
Multimodal Analgesia	Treatment / Therapy	✓	✓		NC			✓		“surgery codes” unspecified		medication names
Multiple Sclerosis	Disease			✓	NC		✓	✓				keywords; regex
Peripheral Arterial Disease	Disease	✓			NC		✓			“concurrent” not fully defined	✓	keywords
Red Blood Cell Indices	Healthy / Valid Data	✓	✓		NC						✓	medication names
Resistant hypertension	Drug Response	✓			NC			✓		“via medication refill data”	✓	medication names (”dose, strength. route, or frequency present”)
Rheumatoid Arthritis	Disease			✓	NC		✓	✓				keywords; regex
Sickle Cell Disease	Disease	✓			NC					“one hospitalization”		
Statins and MACE	Drug Response	✓	✓		NC			✓	✓		✓	medication names; keywords
Steroid Induced Osteonecrosis	Drup Response		✓		NC			✓				keywords
Systemic Lupus	Disease	✓										keywords
Type 2 Diabetes	Disease			✓	NC			✓				keywords; regex
Type 2 Diabetes Mellitus	Disease	✓	✓	✓	XLSX	KNIME; SQL fragments		✓		“self-reported data from a questionnaire”	✓	medication names; regex
Urinary Incontinence	Treatment / Therapy	✓								No value sets provided		executable Python code
Warfarin Dose/Response	Drug Response		✓						✓	“Mention of warfarin (Coumadin) at any time in history”		keywords
White Blood Cell Indices	Healthy / Valid Data	✓			NC					“inpatient hospitalization, ER OR Urgent Care visit”	✓	

Table 2.3: Manually extracted metadata from PheKB.

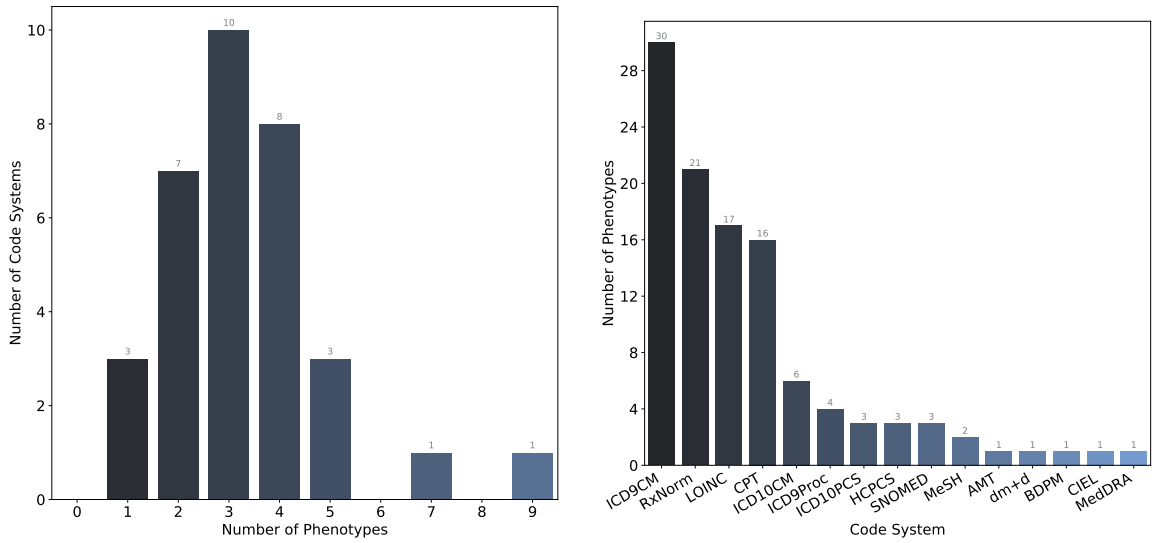
NC – Non-computable (e.g., Word or PDF), **XLSX** – Microsoft Excel file, **CSV** – Comma Separated Value file, **KNIME** – KoNstanz Information MinEr[94] files, **SQL** – Structured Query Language

2.4.4 Terminologies

Figures 2.4 and 2.5 provide histograms related to codes, code systems and value sets. Table 2.4 provides terminology data in tabular form. The table and figures were generated using automated analysis of the ELM representation of the phenotype definitions and the value sets in FHIR format. Most phenotypes (28) use four or fewer code systems, and almost all (30) use ICD-9 codes. RxNorm (21), LOINC® (17), and CPT (16) are the next most commonly used. Five code systems (AMT, dm+d, BDPM, CIEL, and MedDRA) are each only used by a single phenotype. About half of all unique codes (7020) are ICD-9 codes, and about a quarter (4112) are ICD-10 codes. CPT (1221) and RxNorm (699) are the next most common. The total number of codes used varies from 5 (*Warfarin Dose/Response*) to 6865 (*Developmental Language Disorder*), with a median of 147 (mean: 509.2, std: 1206.3). The total number of value sets used ranges from 1 (*Lipids and Sickle Cell Disease*) to 19 (*Resistant Hypertension*), with a median of 5 (mean: 6, std: 4.4).

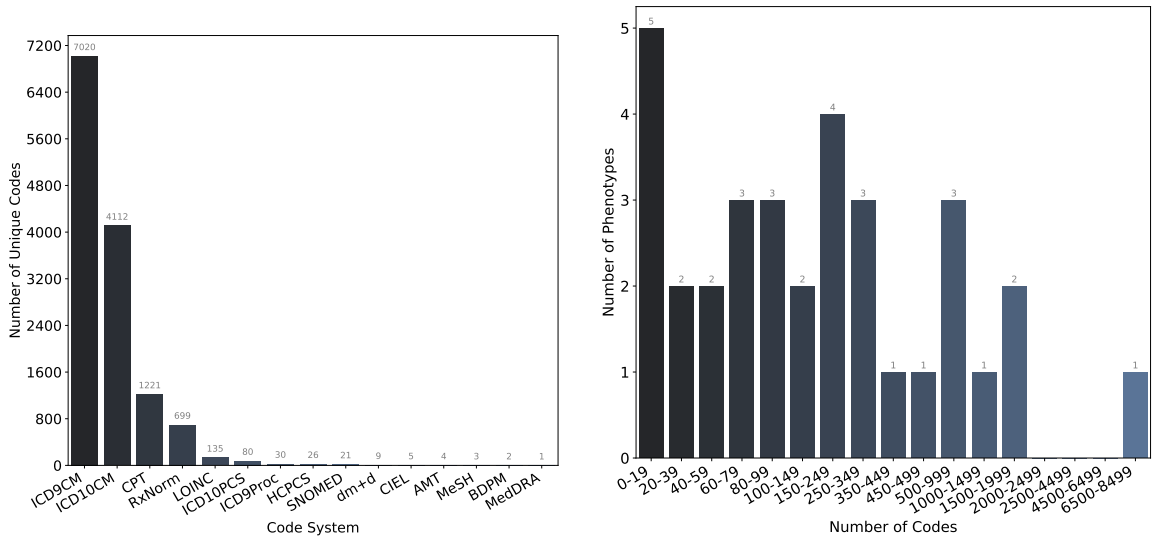
Name	Codes	Code Systems	Value Sets
Asthma Response to Inhaled Steroids	147	2	8
Atrial Fibrillation	192	3	3
Autism	84	1	2
Benign Prostatic Hyperplasia	58	4	5
Bone Scan Utilization	682	5	4
Cardiac Conduction	308	4	11
Cataracts	61	3	3
Clopidogrel Poor Metabolizers	77	3	5
Crohn's Disease	22	2	2
Developmental Language Disorder	6865	2	6
Digital Rectal Exam	176	7	3
Drug Induced Liver Injury	1590	3	9
Familial Hypercholesterolemia	120	4	17
Height	202	3	5
Herpes Zoster	466	3	5
High-Density Lipoproteins	1912	4	15
Hypothyroidism	385	9	8
Lipids	9	1	1
Multimodal Analgesia	564	2	4
Multiple Sclerosis	7	2	3
Peripheral Arterial Disease	1020	4	11
Red Blood Cell Indices	781	4	9
Resistant hypertension	292	3	19
Rheumatoid Arthritis	251	2	4
Sickle Cell Disease	9	1	1
Statins and MACE	93	5	7
Steroid Induced Osteonecrosis	13	2	2
Systemic Lupus	43	3	6
Type 2 Diabetes	85	4	5
Type 2 Diabetes Mellitus	74	3	7
Urinary Incontinence	173	5	2
Warfarin Dose/Response	5	3	3
White Blood Cell Indices	39	4	4

Table 2.4: Total numbers of codes, code systems, and value sets used by each phenotype.



(a) Number of code systems used

(b) Code system utilization



(c) Number of unique codes per system

(d) Total number of codes used

Figure 2.4: Histograms of code and code system usage.

ICD9CM – International Classification of Diseases, Ninth Revision, Clinical Modification, **ICD10CM** – International Classification of Diseases, Tenth Revision, Clinical Modification, **LOINC** – Logical Observation Identifiers Names and Codes, **CPT** – Current Procedural Terminology, **ICD9Proc** – International Classification of Diseases, Ninth Revision, Procedures, **ICD10PCS** – International Classification of Diseases, Tenth Revision, Procedure Coding System, **HCPCS** – Healthcare Common Procedure Coding System, **SNOMED** – Systematized Nomenclature of Medicine, **MeSH** – Medical Subject Headings, **AMT** – Australian Medicines Terminology, **dm+d** – Dictionary of Medicines and Devices, **BDPM** – Public Database of Medications, **CIEL** – Columbia International eHealth Laboratory, **MedDRA** – Medical Dictionary for Regulatory Activities

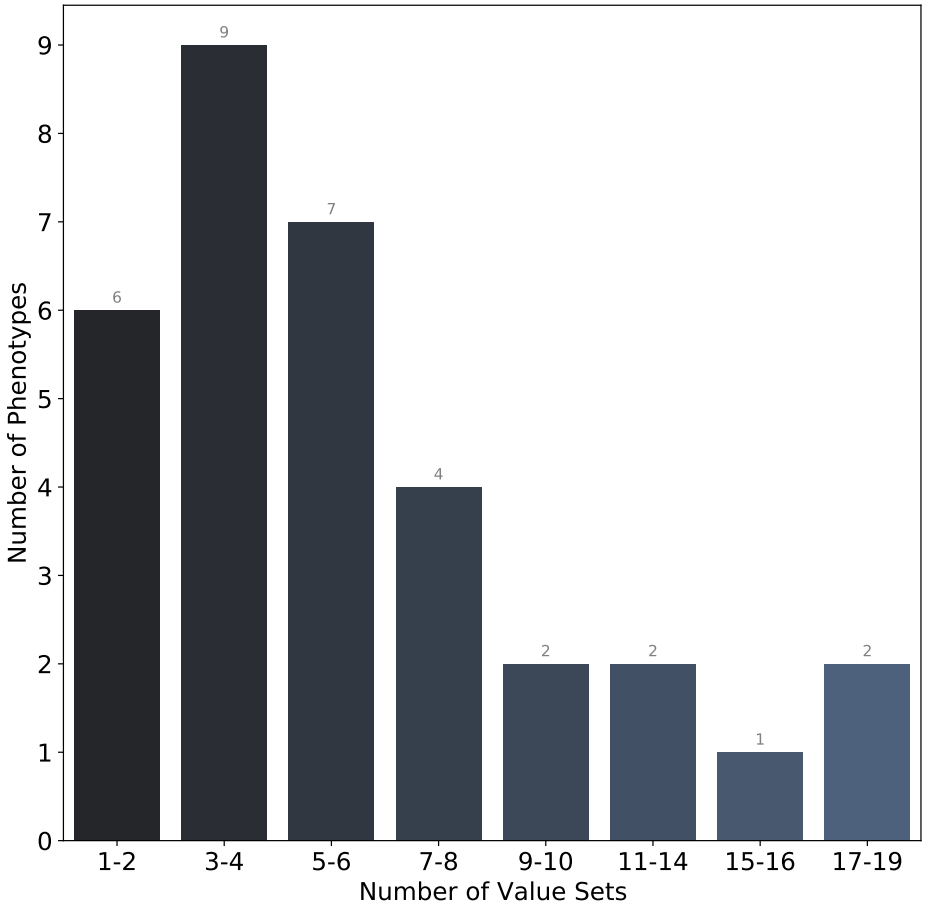


Figure 2.5: Number of value sets used.

2.4.5 Logical Expressions

Table 2.5 lists the total number of expressions used in each phenotype broken down by expression category. Figure 2.6 presents this information visually, and figure 2.7 provides a histogram of the individual expressions within each category. Figure 2.8 illustrates the data types utilized by literal expressions.

The most widely used expression categories are literal (767), data (455), logical (341), and collection (292), with the later three categories used by every phenotype. The least commonly used expression categories are aggregate (30) and arithmetic (80). The total number of expressions used ranges from 6 (*Autism*) to 248 (*Familial Hypercholesterolemia*), with a median of 66 (mean: 76.5, std: 58.8).

Only two types of aggregate expressions were used, with `count` (28) being the most common. The `exists` (201) expression is the most common collection expression used, with `equal` (109) and `if` (141) the most common comparison and conditional expressions respectively. The `query` (226) and `retrieve` (228) data expressions were the most common non-literal expressions overall, while the `aggregate` data expression was used only once. The `and` (170) expression was the most common logical operator, being used about twice as many times as `not` (83) and `or` (88). The `start` (30) operator, which extracts the start date or time from a temporal interval is the most common temporal expression, followed by `in` (29), which checks whether a date or time is in a given interval.

Figure 2.8 shows that terminology literals are the most commonly used types, with the `code`, `code system`, and `concept` types each occurring in 27 or more phenotypes. Next common are primitive types like `Integer` (24) and `Boolean` (20), followed by a long tail of quantities with various units. In total, 17 phenotypes make use of a `Quantity` data type.

Name	Aggregate	Arithmetic	Collection	Comparison	Conditional	Data	Literal	Logical	Temporal	Total
Asthma Response to Inhaled Steroids	0	0	12	10	12	21	44	25	8	132
Atrial Fibrillation	0	0	3	0	0	3	0	3	0	9
Autism	0	0	2	0	0	2	0	2	0	6
Benign Prostatic Hyperplasia	1	0	5	4	1	7	6	7	1	32
Bone Scan Utilization	2	0	6	6	7	12	17	7	8	65
Cardiac Conduction	0	15	15	17	1	26	34	28	9	145
Cataracts	1	0	2	3	1	4	5	3	1	20
Clopidogrel Poor Metabolizers	0	9	8	3	6	16	25	9	5	81
Crohn's Disease	1	0	1	1	0	2	1	1	0	7
Developmental Language Disorder	0	0	6	1	0	6	4	8	1	26
Digital Rectal Exam	0	1	4	4	5	8	16	4	3	45
Drug Induced Liver Injury	0	3	18	16	7	28	36	21	10	139
Familial Hypercholesterolemia	5	10	27	34	14	44	81	27	6	248
Height	0	0	6	13	10	13	23	9	8	82
Herpes Zoster	1	2	6	6	5	13	17	11	5	66
High-Density Lipoproteins	4	0	14	26	14	35	75	9	10	187
Hypothyroidism	0	3	13	2	8	15	12	12	5	70
Lipids	0	0	6	3	5	6	11	2	3	36
Multimodal Analgesia	0	0	10	12	5	12	49	6	0	94
Multiple Sclerosis	0	0	3	0	0	3	0	2	0	8
Peripheral Arterial Disease	2	0	10	7	9	15	20	9	1	73
Red Blood Cell Indices	0	12	13	6	9	21	30	14	11	116
Resistant hypertension	1	15	23	19	10	34	66	18	10	196
Rheumatoid Arthritis	0	0	6	0	0	6	3	7	0	22
Sickle Cell Disease	2	0	3	4	7	7	15	4	2	44
Statins and MACE	2	1	8	11	7	18	27	12	9	95
Steroid Induced Osteonecrosis	1	2	7	19	11	23	52	14	2	131
Systemic Lupus	1	1	5	4	2	7	11	6	0	37
Type 2 Diabetes	0	0	10	4	1	10	14	11	1	51
Type 2 Diabetes Mellitus	4	0	18	10	3	16	24	30	0	105
Urinary Incontinence	2	0	5	6	7	8	17	6	4	55
Warfarin Dose/Response	0	6	13	6	9	10	32	8	5	89
White Blood Cell Indices	0	0	4	0	0	4	0	6	0	14
Total	30	80	292	257	176	455	767	341	128	

Table 2.5: Expression counts per category.

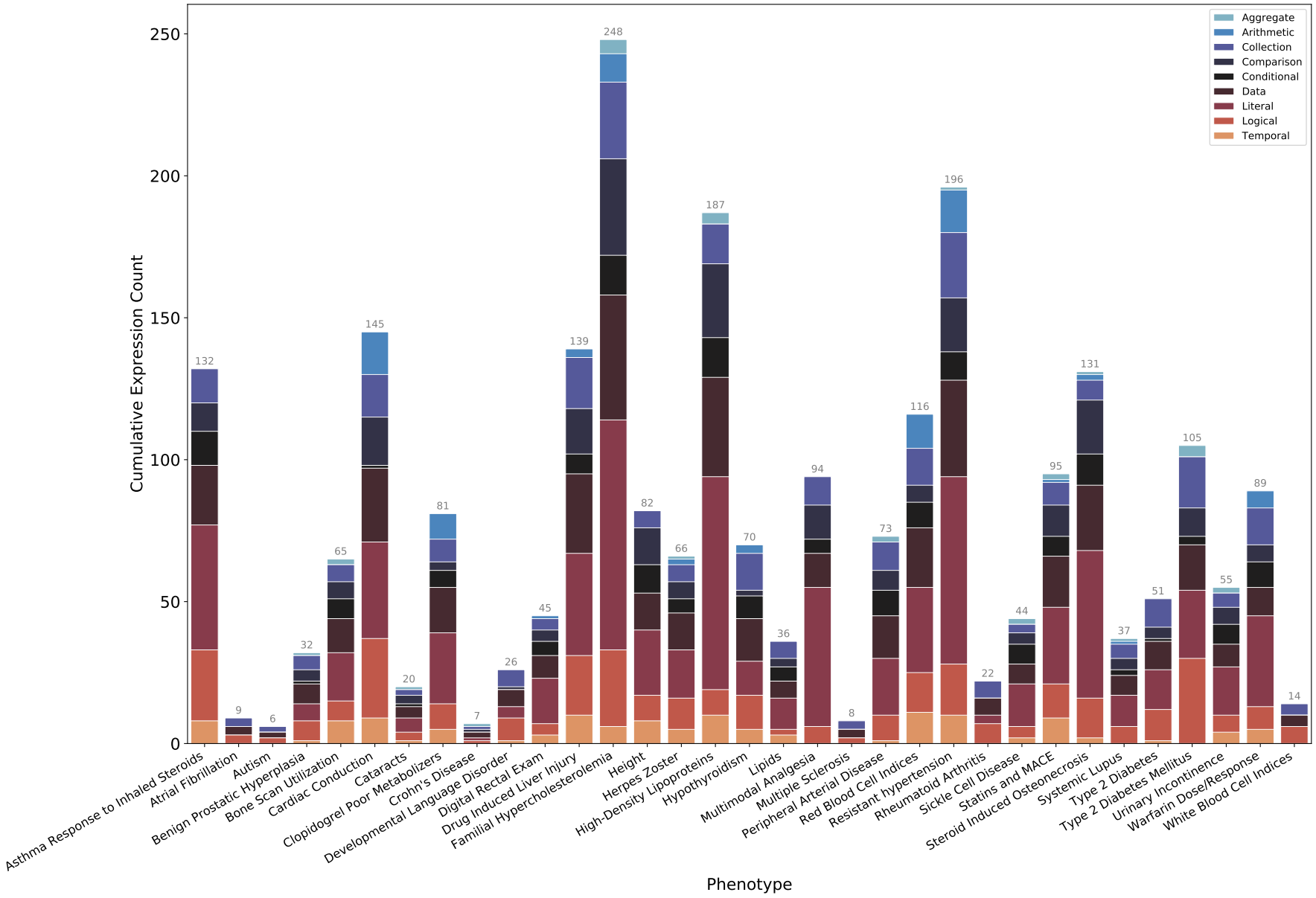


Figure 2.6: Expressions used by all phenotypes.

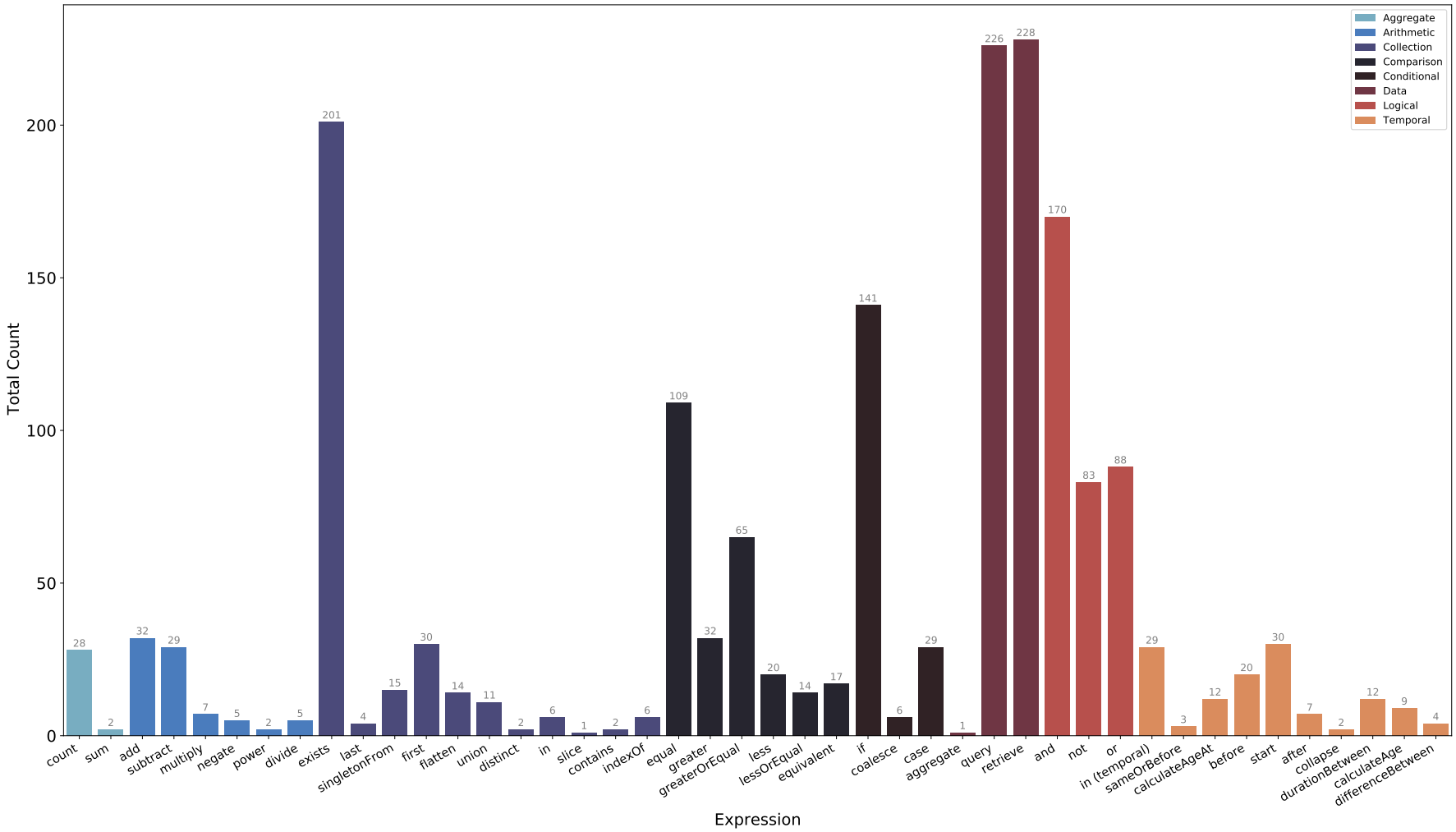


Figure 2.7: Total numbers of individual expression types by category excluding literal expressions and including helper libraries.

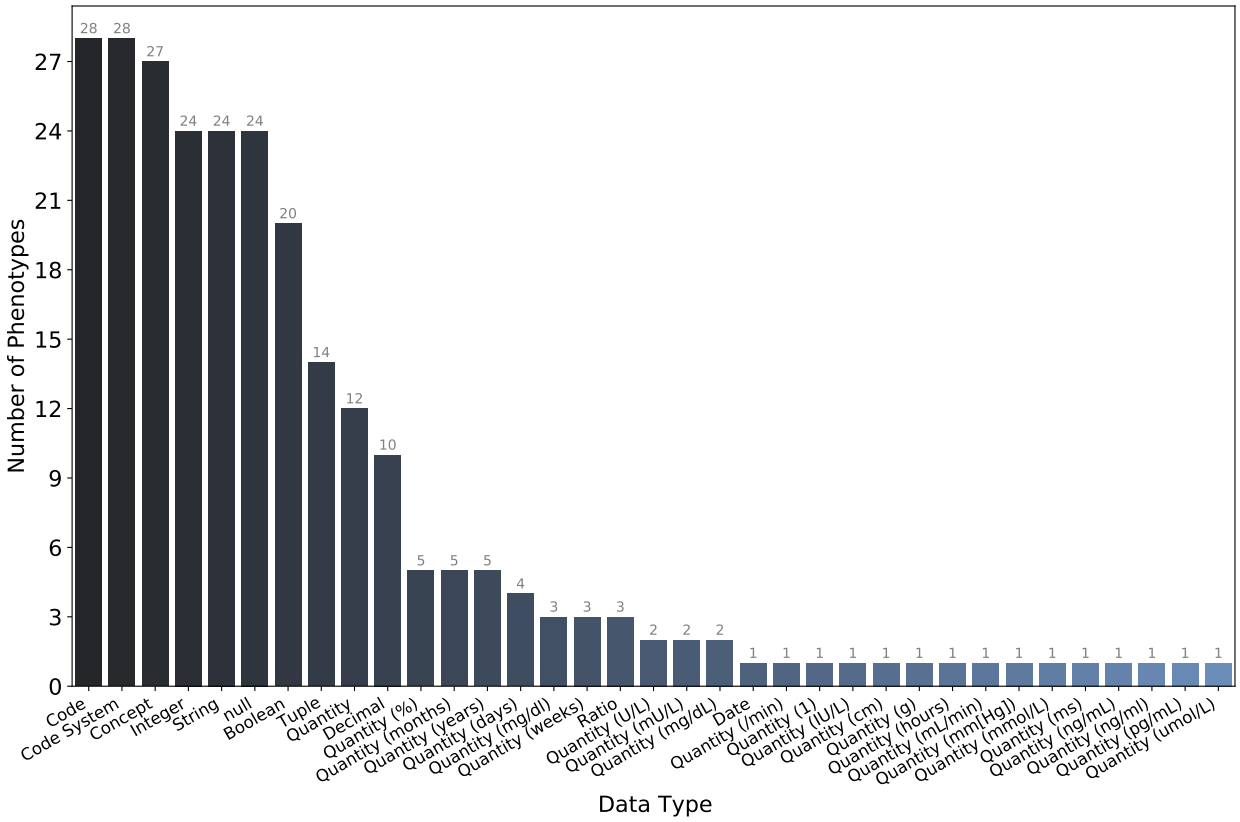
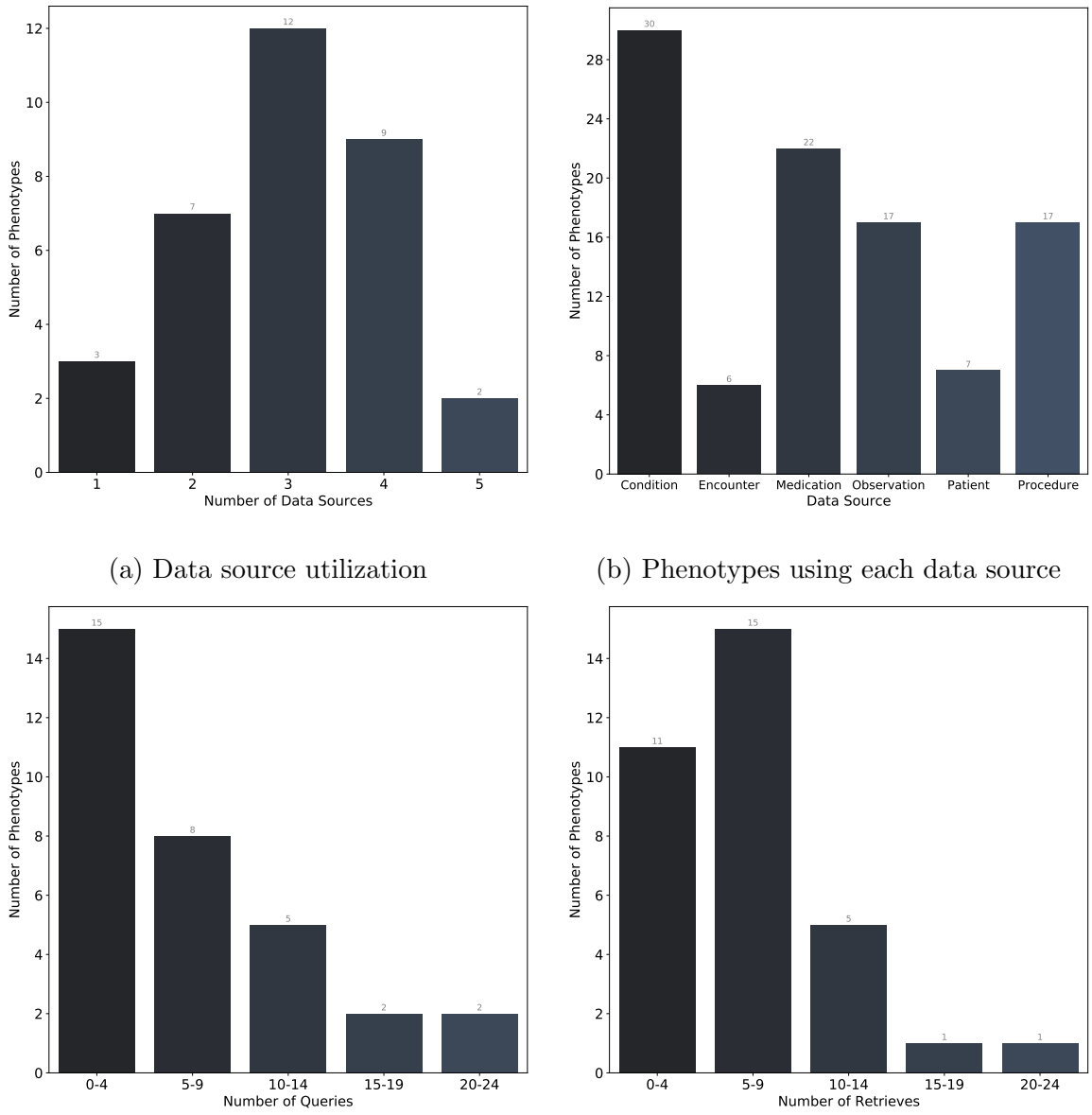


Figure 2.8: Utilization of various data types.

2.4.6 Data Sources

Figure 2.9 illustrates how many data sources are used per phenotype definition and how many phenotypes used each data source. Also provided are histograms of the numbers of `query` operations (used to filter and shape collections of data) and `retrieve` operations (used to fetch records for data sources). The majority of phenotypes (22) used 3 data sources or fewer. Conditions were the most common data source, used by almost all (30) phenotypes, followed by medications (22), procedures (17), and observations (17). Demographic and encounter data were the least frequently used.



(a) Data source utilization

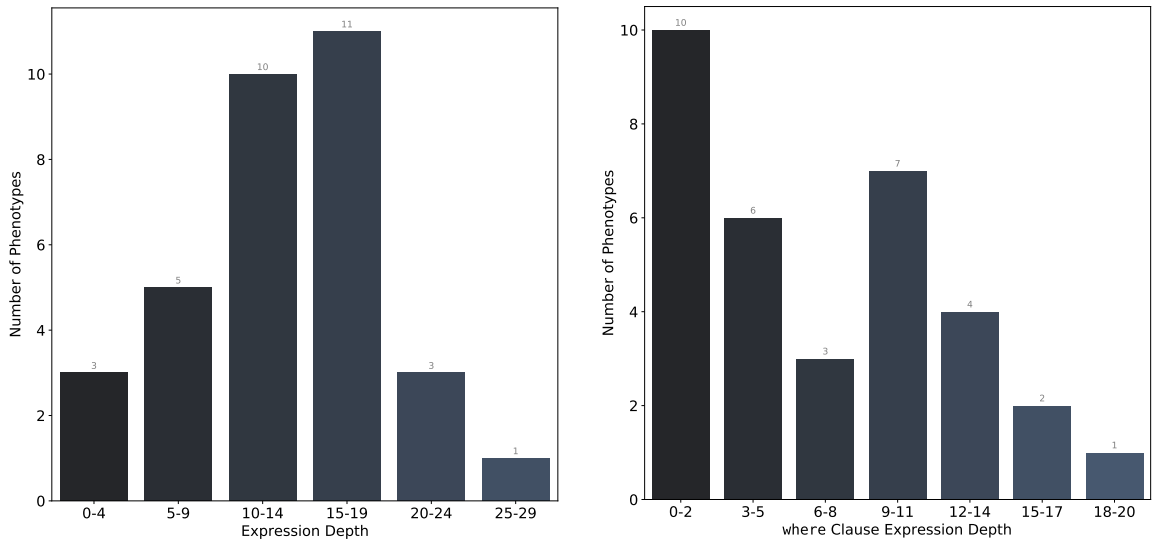
(b) Phenotypes using each data source

(c) Query utilization

(d) Retrieve utilization

Figure 2.9: Histograms of data sources and expressions.

The majority of phenotypes use fewer than ten `query` and `retrieve` expressions. The outliers are *Familial Hypercholerolemia* (20 queries and 24 retrieves), *Resistant Hypertension* (26 queries and 7 retrieves), and *High-Density Lipoproteins* (17 queries and 18 retrieves).



(a) Total expression depth

(b) `where` clause expression depth

Figure 2.10: Total and `where` clause expression depths.

2.4.7 Expression Depths

Expression depths indicate how many logical expressions are applicable concurrently, which is roughly correlated with how many phenotype definition criteria are concurrently applicable. Figure 2.10 provides a histogram of total expression depth as well as `where` clause expression depth. `where` clause expression depth is an indicator of how complicated data filtering expressions are. Finally, figure 2.11 illustrates expression depth per expression category, which shows how many expressions of each different category are concurrently applicable.

The lowest total expression depth is 4 (*Multiple Sclerosis*, *Crohn's Disease*, and *Autism*) and the highest is 27 (*Familial Hypercholesterolemia*), with a median of 14 (mean: 13.5, std: 5.7). Seven phenotypes have a `where` clause expression depth of zero (*White Blood Cell Indices*, *Rheumatoid Arthritis*, *Multiple Sclerosis*, *Crohn's Disease*, *Atrial Fibrillation*, *Autism*, and *Developmental Language Disorder*), meaning that data is only filtered by value

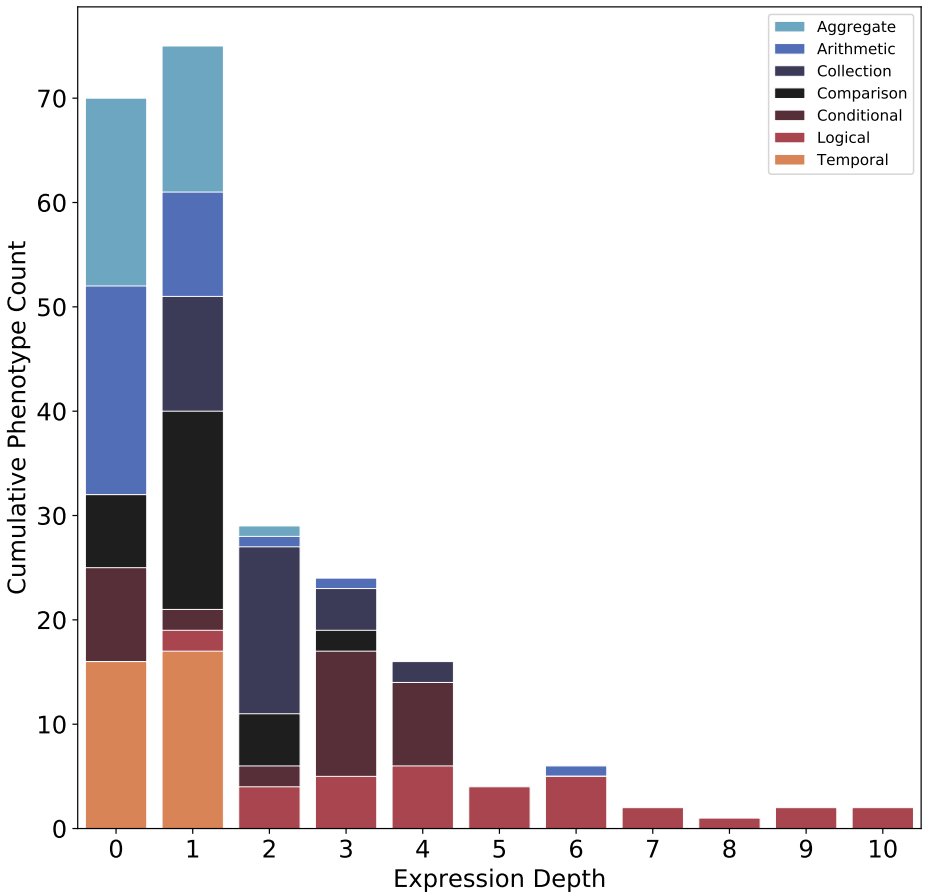


Figure 2.11: Depths per expression category.

set and no other criteria. *Red Blood Cell Indices* has a the highest `where` clause expression depth (18), and the median `where` clause expression depth is 7 (mean: 6.6, std: 5.4).

Many phenotypes have expression depths of zero or one for aggregate, arithmetic, collection, comparison, and conditional expressions. Logical expressions always have a depth of at least one. There are some instances of expression depths in the two to four range for conditional, collection, comparison, and logical expressions, but only logical and arithmetic expressions have a depth of five or greater. Only logical expressions have a depth greater than six, with a maximum of 10 in two cases (*Red Blood Cell Indices* and *Familial Hypercholesterolemia*).

2.5 Discussion

The artifacts that comprised the phenotype definitions can be divided into two high-level categories: logic and tabular data. The tabular data consists of value sets of codes from various code systems and lists of keywords or regular expressions used for NLP, although we did not analyze the latter in depth. Phenotype logic can be further divided into two categories: clinical logic and operational logic. Clinical logic is the core of the phenotype definition, and describes the clinical definition of the phenotype. Clinical logic includes which diagnoses are relevant, which procedures, medications, and lab orders are associated with the phenotype, as well as patient demographic criteria that should be considered. Operational logic is also important, and while it contributes to the clinical definition, it is typically a bridge to how data is recorded in the EHR. Although some patterns have been observed[95], operational logic can be difficult to express accurately using universally applicable logical expressions. For example, the *Herpes Zoster* phenotype requires that a matching patient have at least 5 years of continuous enrollment. The reason for this requirement is to "increase the probability that a subject's status with respect to herpes zoster infection is known by the health care system." This does not necessarily increase the correctness of the phenotype definition, but may nevertheless increase the positive predictive value. Another very common operational criterion is the requirement that a patient have at least 2 diagnoses of a given condition. This criterion is relatively simple to define using universally applicable CQL logic, while the concept of enrollment is determined differently at different institutions. One solution to this problem, which is available when using a modular formal representation, is to have local implementations for common operational criteria that are used during phenotype

execution. This is the same approach used in computer software, where system libraries provide routines with known names and well-defined parameters, but the implementation varies according to the operating system.

PheKB was initially created by the eMERGE Network, which is primarily focused on conducting research that combines genetic data with EHR data. For this reason, the majority of phenotype definitions we reviewed are written in a way that supports this type of research. Specifically, covariates of interest for the relevant research study are often referenced in the phenotype, even if they do not form part of the phenotype definition itself. This is likely because phenotyping and covariate extraction have historically been done simultaneously. However, this can make it more difficult to interpret the phenotype definition, especially when research study instructions are given directly inline with the phenotype logical criteria. For example, in the *White Blood Cell Indices* phenotype, the definition requires that the implementer flag patients with Alzheimer's disease. This information is not used for inclusion or exclusion purposes, but is preparatory work for the genome-wide association study (GWAS) that the phenotype was developed for. Because of the history and purpose of PheKB, the conflation of phenotype definition and research data dictionaries is to be expected, and was probably beneficial to the initial users of PheKB. However, for phenotype definition clarity and re-usability purposes, it is likely better to completely separate research-specific data preparation steps from the phenotype definition.

During this work we experienced first hand many of the challenges that face phenotype implementers. We encountered numerous occurrences of ambiguity, underspecificy, and imprecise language. For example, the *Clopidogrel Poor Metabolizers* phenotype uses the phrase "within 30 days", but does not specify whether the interval boundary should be inclusive

or not, or whether the 30 days both before *and* after the event should be considered. In each case the primary CQL developer had to confer with colleagues in order to determine the exact semantics of the phenotype definition. Even then, we would occasionally rely on subjective decisions regarding the intent. This resulted in a considerable slow down in implementation, and the resulting implementation may still be erroneous. The only certain way to determine semantic correctness would be to reach out to the original phenotype definition authors, who may not have a definitive answer (given elapsed time from when some phenotypes were authored). Additionally, some phenotype definitions relied heavily on domain or tribal knowledge not specified within the definition itself. This makes it difficult for non-clinicians or healthcare outsiders to replicate research or use existing phenotype definitions for new research. For example, the *High-Density Lipoproteins* phenotype requires that a cohort member have at least one "random glucose test", but does not specify how these tests are to be identified. We also encountered contradictory criteria definitions, for example, the *Bone Scan Utilization* phenotype requires that a cohort member be both > 35 and ≥ 35 years old. A formal representation may not eliminate all these issues, but it would require phenotype authors to be more precise at the definition phase, which would reduce the cognitive load on implementers.

The self-reported demographic metadata is not very complete (see table 2.2), and where this metadata is provided, the same information rarely forms part of the phenotype definition itself. The exceptions are age and gender, which do usually form part of the phenotype when clinically relevant. The self-reported data modalities are very complete, with only one phenotype not reporting anything. Further, the self-reported data modalities match what we found in our computational analysis, namely that ICD-9 codes are the most common,

followed by medications and lab values (which we modeled using LOINC® codes). However, the number of categories in the self-reported data are limited and don't allow authors to specify if they use encounter data, observations (besides vitals and laboratories), or demographic data. The self-reported number of data modalities used is 4 or fewer in most cases, while we found it to be 3 or fewer in most cases. Since we excluded the very commonly used NLP modality, this discrepancy makes sense. Due to the incompleteness and limited categories of self-reported metadata, this information is likely not very useful to implementers, who must still read phenotype definitions in full to determine implementation requirements. Using a formal representation can mitigate this problem by enabling the automated extraction of complete, detailed, and accurate metadata.

The prevalence of certain expressions such as data retrieval and manipulation expressions as well as literal expressions may be expected. What may be surprising is the relatively low usage of arithmetic and aggregate expressions, which indicates that in most cases data values are used directly, rather than to construct derived values. Both the `count` and `sum` aggregate expressions are used as cardinality constraints (e.g., at least 2 diagnoses required) and not in an arithmetic context. The highly used existential operator (`exists`) is used for the same purpose (e.g., does an observation exist that meets certain requirements). Additionally, even though about two thirds of phenotypes use temporal expressions, the total number used is relatively small. The fact that the `and` operator is used about twice as much as the `or` and `not` operators indicates that the conjunction of criteria is much more common than their disjunction or negation. This makes sense, since conceptually, phenotypes are defined by clusters of concurrent criteria rather than by the disjunction of many different criteria.

Overall expression depths seem to be normally distributed around 15, while `where` clause

expression depths appear to be bimodal, but generally trend down at high values. The downward trend implies that complicated data filtering expressions are uncommon. Most expression categories are not very deeply nested, and only logical expressions (and in one case arithmetic expressions) have a depth of 5 or higher. This can be interpreted to mean that conceptually simple criteria are combined in complex ways using `and` and `or` expressions. This is confirmed by the flowcharts provided with some phenotype definitions, which may have a complicated topology, but the criteria represented by each node are relatively simple.

In general, the total number of expressions per phenotype is not very high, with a median of just 66. The total number of expression types is also quite low, at just 44 (CQL has over 200 expression types). There are several possible reasons for the simplicity of the phenotypes in the data set. First, in our experience, implementing even simple phenotype definitions is quite challenging, so authors may choose to keep definitions simple to make implementation practical. Second, the phenotypes generally restrict themselves to data available in the EHR, which may be simplified, as some of this data is for billing purposes. Further, since the PheKB phenotypes are designed to be shared, authors may limit themselves only to data available to most implementers, such as the most basic data elements. Third, since phenotypes are created as narrative text, the lack of a formal expression language may be a factor that limits the level of detail provided. Finally, related to the previous point, it may be the case that since we used CQL to represent the phenotype definitions, the resulting number of expressions is low, as complex clinical and temporal operations are available as single expressions in this language.

There is no clear correlation between the severity or complexity of presentation of a disease and the number of expressions used. For example, something ostensibly simple like

Height has ten times as many expressions as *Multiple Sclerosis*. There are also two *Type 2 Diabetes* phenotypes, one with 51 expressions and one with 105, so even the same disease can be represented in vastly different ways. This implies that a large part of phenotype definition complexity is determined by the level of detail that the author decided to use. This is independent of any formal representation, and may depend on the intended use of the phenotype definition.

While it was not the objective of this study to evaluate FHIR and CQL as a standards-based representation, we provide a short discussion here of the insights gleaned by using this approach. One benefit of using a phenotype representation based on an already widely used standard is the existence of tools for working with the standard. Another important factor to consider is verifiability through testing. FHIR provides the `TestScript` resource that can be used to define and validate specific behavior, and CQL has the CQL Testing Framework (shown in figure 2.1) that can be used to provide confidence in the semantic correctness of phenotype definitions. Other advantages of using mature standards include the availability of documentation and expertise.

There are additional benefits to using healthcare-specific standards, as they usually provide useful domain-specific conceptual models. For example, FHIR provides resources for modeling clinical entities, attributes, and relations. An important conceptual model used by CQL is the decision to have both human and machine-readable versions of logical expressions, which at the same time minimizes the learning curve and reduces effort required to implement a CQL engine, since a CQL to ELM parser is provided by the community.

We observed that many phenotypes share similar logical units. For example, finding the most recent value for a lab test or other observation, or checking the age of the patient

at the oldest or newest occurrence of a diagnosis code. Any formal representation should therefore have the ability to represent shared logic. Due to the modularity of CQL, we were able to extract shared logical units into separate libraries that can be accessed by multiple phenotype definitions. This same mechanism can also be used to express localized operational logic, provided interfaces are first agreed upon, or customized clinical logic based on local guidelines and processes. Examples of local customization include using different terminologies (e.g. SNOMED vs LOINC®), using different value sets to encode a specific condition, or using different thresholds or different sub-rules for defining phenotypes. While standards can help reduce the need for local customization, it seems unlikely that it will be possible to completely eliminate this need, so any formal standard must support this use case.

The CQL and FHIR-based representation used in this work supports local customization in a transparent and well defined manner. To support different terminologies or value sets, the implementer simply needs to ensure that the referenced value set is updated to reflect the local context. The advantage of this approach is that the CQL library itself does not need to be updated at all, only the `ValueSet` resource. In addition, this `ValueSet` resource only needs to be updated once and the update will be reflected in all the phenotype definitions that reference this value set. Additionally, local customization of phenotype logic can be achieved by extracting customizable logic into CQL libraries that can be replaced in local implementations. Finally, since CQL is data model independent, sites that do not use the same data model as a given CQL library have several options. These sites could implement translation logic that transforms the machine-readable ELM into a new ELM referencing the preferred data model, or they could implement a new CQL engine that performs this

translation transparently. While these options incur an upfront implementation cost, it will only be paid once.

2.5.1 Limitations

We note the following limitations in this work. First, our data set is relatively small, consisting of only 33 phenotype definitions, and while we believe we have reached sufficient saturation required for our analysis purposes, there may exist additional phenotype definitions that would alter our conclusions. We recognize that the two primary authors may have missed subtleties in some definitions during translation, although this was mitigated in part by the independent review of each phenotype by the other author, and consulting additional colleagues where ambiguities needed to be discussed.

While we took steps to ensure that our implementations were as simple as possible while remaining semantically correct, alternative implementations may be possible since CQL is a highly expressive language. This could impact our calculated measures, but since all implementations were reviewed by both authors, there should be some degree of consistency, which means that comparisons between implementations are still informative. An additional limitation is that our CQL-based representations were not clinically validated. Therefore, even though we selected only clinical valid phenotype definitions to translate, our resulting representations have not themselves been clinically validated.

We also note that all the definitions we translated were designed to detect patients with a single condition. Even though EHR data provides the opportunity to conduct research on patients with complex comorbidities, the phenotype definitions in our data set were not designed to identify cohorts of such complex patients. The phenotypes analyzed are

reminiscent of those that might be used for recruiting patients for randomized controlled trials, but we hope this work will provide some insights that lead to methods for developing higher fidelity phenotype definitions, and that these definitions can be used for so-called *deep phenotyping*.

The decision not to include NLP directives in our analysis was purposeful, but impacts any conclusions we may draw about individual phenotypes. We know that most (28) phenotype definitions make use of some form of NLP, so we are omitting data from a significant number of definitions. However, almost all definitions including NLP directives simply provide a keyword list or regular expressions, and not higher-level entities, negation, or temporal constructs. In some cases, no details are given about how the NLP should be implemented. For example, the *Red Blood Cell Indices* phenotype definition says only "NLP was implemented to that regard" (referring to identifying patients taking specific medications). So, we believe that including a more detailed analysis of NLP data elements would not be informative due to their underspecificity in the dataset.

Despite the above limitations, we have characterized a data set of clinically validated phenotype definitions along several dimensions, and have presented both the raw data and our analysis. We have additionally identified the essential components of any phenotype definition, and highlighted some important requirements that any potential formal representation should fulfil.

2.6 Conclusion

Phenotype definitions consisting only of narrative descriptions and other unstructured artifacts present significant implementation challenges. These implementation challenges are a rate limiting factor in the generation of biomedical knowledge. In this work we characterize the nature of a set of clinically validated phenotype definitions, and propose some requirements for a formal representation. Such a formal representation may reduce ambiguity, imprecision, and contradictions, as well as decrease implementation time by facilitating automation. These improvements may increase the velocity at which observational research can be conducted.

The phenotypes analyzed show significant variation in small details, but are all composed of the same high-level components, namely tabular data and logical expressions. The most important type of tabular data analyzed here is value sets, which can readily be represented in a standard format. Logic can be clinical or operational and both can be represented in a standard format, however, operational logic may in some cases be highly localized. Therefore, any standard representation must support the seamless combination of local and universal logic.

There are numerous advantages to using popular healthcare-specific standards, such as convenient conceptual models, mechanisms for verification, and the availability of tools, documentation and expertise. Using a logical representation that is data model independent may facilitate automated execution by allowing implementation sites to implement their own data providers for existing phenotype definitions. Similarly, the use of common standard terminologies may enable automatic mapping during local execution.

A significant amount of data useful for EHR-driven phenotyping may be stored in clinical notes. The body of research focused on extracting this data is growing, but to our knowledge, no widely accepted standard representation for NLP metadata and processes has yet emerged. The PhEMA research team is working on methods for integrating NLP into both FHIR[27] and CQL[38], and in future work we hope to integrate this research into our standards-based phenotype representation.

Finally, we hope to continue to investigate whether FHIR and CQL are viable representations by using these representations in clinical research. We have conducted early work that provides evidence that these standards may be suitable[37], and we have further work in this regard underway.

CHAPTER 3

TOWARD CROSS-PLATFORM ELECTRONIC HEALTH RECORD-DRIVEN PHENOTYPING USING CLINICAL QUALITY LANGUAGE

3.1 Introduction

Learning Healthcare Systems (LHS) are organizations in which the delivery of care generates data and insights that can be analyzed and transformed into biomedical knowledge. This knowledge can then be used to improve the quality and efficacy of healthcare[3]. A core aspect of generating this knowledge is the identification of patient cohorts in the electronic health record (EHR) matching certain clinical criteria, a process commonly referred to as EHR-driven phenotyping. EHR-driven phenotyping has applications across the continuum of LHS to conduct case-control and cohort studies, clinical trial recruitment, clinical decision support (CDS), and quality measurement[8].

We have established the Phenotype Execution and Modeling Architecture (PhEMA), an open-source infrastructure to support clinicians, researchers, informaticists, and data analysts in standards-based authoring, sharing, and execution of computable phenotype definitions[96]. In this work we continue to improve the PhEMA tools by proposing to adopt Clinical Quality Language (CQL)[97], a Health Level Seven International (HL7) standard for formally representing logical expressions, as the computable phenotype representation.

Our hypothesis is that if a standards-based phenotype representation is used, it will enable execution across data platforms with a one-time cost. That one-time cost is the development of a CQL engine for each target platform, and this cost is preferable to manual phenotype translation, as it ultimately enables cross-platform phenotyping at scale. We investigate whether this approach does enable cross-platform phenotyping and demonstrate a newly built CQL evaluation engine that is able to execute CQL phenotype definitions against the Observational Health Data Sciences and Informatics (OHDSI) platform[18].

We used a clinically validated phenotype definition for patients with heart failure (HF), a common, costly, and morbid condition affecting over 6 million US adults and a high public health priority[98]. The system was validated at multiple institutions and across data platforms, and is made available on GitHub to complement the current set of tools used by the observational research community, with the hope that our methods will contribute towards the future convergence of phenotyping systems.

3.2 Background

In general, EHR-driven phenotyping is a two-step process: 1) defining the phenotype and 2) executing the phenotype. First, a phenotype definition must be developed, which is a resource-intensive process involving multi-disciplinary teams, and often requiring several iterations to produce a high quality, clinically valid result[21,35]. Phenotype definitions typically consists of (i) clinical data elements of interest, such as demographics, medications, diagnoses, encounters, laboratory results, and other clinical observations, (ii) lists of codes from published terminologies, called *value sets*, and (iii) Boolean, aggregate, and temporal logical

expressions that relate the data elements and value sets (phenotype *logic*). Additionally, the phenotype definition must be validated against a gold standard, most often derived from a resource-intensive manual chart review[23,34].

Second, in order to assemble the cohort of interest, the phenotype definition must be *executed* against a clinical database. Without a directly executable standard representation, this involves human interpretation of a narrative description or flowchart illustrating the phenotype definition and translation into machine-executable code, such as SQL or R. This is a time-consuming and error-prone process, which sometimes involves translating value sets into local terminologies[45,99]. Such phenotype definitions are not portable or scalable, as these steps must be repeated at every implementation site, resulting in duplication of effort and highly variable results.

In contrast, *computable* phenotype definitions are represented in an unambiguous formal language and can be executed against a database with minimal human intervention, reducing implementation effort and variability, increasing transparency, and enabling high-throughput phenotyping[16]. Two approaches enable computable phenotype definitions: common data models (CDMs) and dedicated phenotype logic execution environments. CDMs allow writing executable code that can be used against different clinical databases without code modifications. Research networks such as the OHDSI network, the Health Care Systems Research Network (HCSRN)[100], Sentinel[101], the electronic Medical Records and Genomics (eMERGE) Network[12–14], the National Patient-Centered Clinical Research Network (PCORnet))[31], and the Accrual to Clinical Trials (ACT) Network[102], have used this approach with much success[28,103]. However, no single CDM is ubiquitous, and the code written for any given CDM is not executable against a different CDM. For example, the PCORnet CDM

and the Observational Medical Outcomes Partnership (OMOP) CDM used by OHDSI both represent similar categories of medical data, however a query written against one cannot be directly executed against the other without modification because the schemas are different.

Logic representation standards like the healthcare-focused Health Quality Measure Format (HQMF) and CDS Knowledge Artifact Specification (KAS), and general logic execution environments such as JBoss® Drools and the Konstanz Information Miner (KNIME) have also been shown to work in select use cases[24,104]. Software code is not based on any formal healthcare-related standard, and while HQMF and CDS Knowledge Artifact Specification (KAS) show promise, they do not have human-readable representations. General logic execution environments may present a significant implementation burden, with some institutions spending significant valuable resources and time, and still failing to get the systems running[19].

Instead, clinicians, informaticists, and data analysts need an approach that allows them to collaborate with institutions using a variety of CDMs, and minimizes the number of times a phenotype has to be written. CQL is a formal logical expression language that supersedes HQMF and CDS KAS, and is intended to be used for electronic clinical quality measures (eCQMs) and CDS, as well as more general clinical knowledge representation use cases. The Centers for Medicare & Medicaid Services (CMS) and HEDIS® (The Healthcare Effectiveness Data and Information Set) have published eCQMs using CQL. The emerging Fast Healthcare Interoperability Resources (FHIR) standard has also adopted CQL as one of its standard logical expression languages[105]. Additionally, there are several tools for authoring knowledge content in CQL, such as the CMS Measure Authoring Tool (MAT)[106] and the Agency for Healthcare Research and Quality's (AHRQ) CDS Connect authoring

tool[107].

CQL is organized into *libraries* – comparable to programming packages – which have the added benefit of being reusable across multiple eCQM and CDS. CQL has both a high level, human-readable representation, and an equivalent machine-readable representation, called the Expression Logical Model (ELM). The ELM is an Abstract Syntax Tree (AST)[52] representation of the language and has both a JSON and XML format. The intention of the language authors is that engine developers should focus on the evaluation of core logic expressed in the ELM, and use existing tools for parsing, expression simplification, and semantic analysis[97]. Furthermore, CQL is data model agnostic, meaning that different data models, such as FHIR, OMOP or the Quality Data Model (QDM), may be utilized with the same logical constructs.

3.3 Methods

3.3.1 Phenotype Selection & Translation

We adopted a pre-existing HF phenotype definition that has been executed and clinically validated against multiple EHRs and sites[108–110]. The HF phenotype definition uses several different data modalities, including demographic data, clinical diagnoses, clinical encounter types, as well as procedure orders. It also uses Boolean logic, temporal logic in the form of patient age and co-occurrence of diagnosis with encounters, and an aggregate function. Additionally, it references a number of common clinical terminologies, including the International Classification of Diseases versions 9 (ICD-9) and 10 (ICD-10), the Current Procedural Terminology (CPT), as well as the Systematized Nomenclature of Medicine (SNOMED).

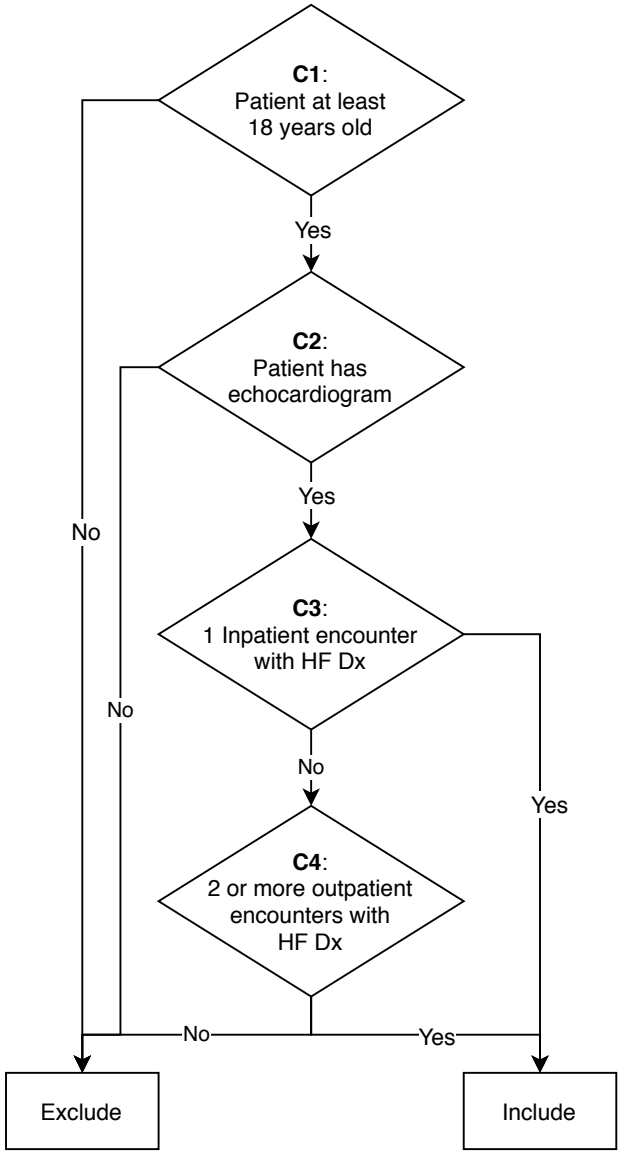


Figure 3.1: The HF phenotype definition. All criteria are labeled **C1** through **C4**.

We began by representing the HF phenotype definition as a CQL library. We selected the FHIR data model for data element references because mappings already exist from the FHIR data model to many popular CDMs[46], and many CQL engine implementations support FHIR[111].

The HF phenotype logic is shown in figure 3.1. Criteria **C1** and **C2** are mandatory, and the patient must also match either **C3** or **C4** to be considered a case. CQL is sufficiently

expressive to represent these criteria, and the source, available in the project's GitHub repository[112] has six total statements. One for each criterion, one to represent the disjunction of **C3** and **C4** (**C***), and one to represent the final conjunction: **C1** `AND` **C2** `AND` **C***.

Two value sets were needed, one for the HF diagnosis codes (Dx VS), which came from three different terminologies (ICD-9, ICD-10, and SNOMED), and one for the echocardiography procedure codes (Echo VS), from the CPT terminology. We used an existing Dx VS from the Value Set Authority Center (VSAC)[47], which is also used by CMS for their HF eCQMs. We created and published a new Echo VS in VSAC. For inpatient and outpatient encounter types, we used individual codes from the ActCode[113] terminology, as recommended by the FHIR standard[114].

3.3.2 CQL Engine Development

We chose to develop a CQL engine, called *CQL on OMOP* (figure 3.2 box 1), for the OHDSI data platform. In addition to its use of the OMOP CDM, OHDSI has existing phenotype definition analysis and visualization tools built upon a Web application programming interface (API), making it possible to validate our results using independent methods. The OHDSI platform represents phenotype definitions in a transportable JSON format, and executes them using a library called *Circe*[115] that provides entities for representing phenotype logic, for example, `CriteriaGroup` and `DemographicCriteria`. CQL on OMOP translates a CQL-based phenotype definition into the Circe representation and then uses the OHDSI Web API to generate the cohort (figure 3.2 box 2(b)).

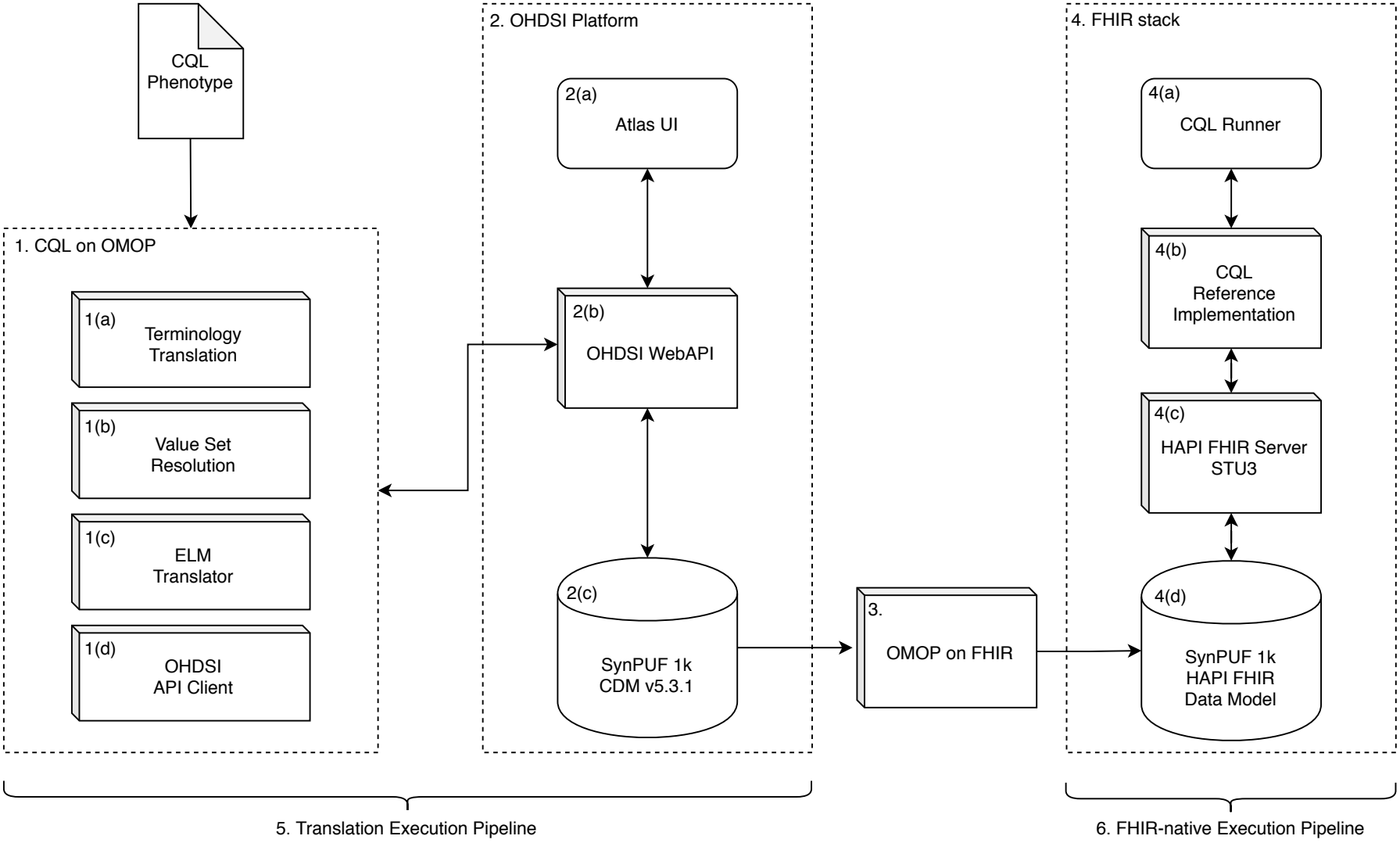


Figure 3.2: Box (1) shows the developed CQL on OMOP engine, box (2) the OHDSI data platform, box (3) the OMOP on FHIR data transformation tool, and box (4) the FHIR-native stack used for cross-platform validation. Box (1) shows our newly developed software, while boxes (2) to (4) are existing systems we leveraged. Pipelines (5) and (6) show the two validation methods. Both NM and WCM used the pipeline (5) architecture with their own data for phenotype execution.

The engine was developed as an open-source Java application[116] and uses libraries provided by the CQL authors to parse CQL and generate an ELM tree[117]. Entities from the Circe library are used to represent cohort criteria in the format expected by the OHDSI Web API. CQL on OMOP runs as a standalone application and can be configured to connect to an instance of the OHDSI Web API. Both CQL and ELM are supported as inputs, as well as value sets in several different formats, including the format produced by VSAC. Finally, the tool is developed in a modular way that makes it easy to add new CQL language features and support new data element correlations.

The process of value set resolution leverages CSV files downloaded from VSAC that were packaged with the HF CQL. The process of resolution (figure 3.2 box 1(b)) matched value set object identifiers (OIDs) within these files and the CQL code. CQL on OMOP used the OHDSI Web API to identify the relevant concepts from each value set and build the OHDSI concept set (figure 3.2 box 1(d)). The OHDSI platform primarily makes use of standard terminologies, with one exception being internal codes defined by OHDSI for visit types. This required us to implement a simple terminology translator between FHIR encounter types and OHDSI visit types (figure 3.2 box 1(a)).

The core contribution of the CQL on OMOP engine is the ELM logic translator (figure 3.2 box 1(c)). The engine implements a rule-based, recursive descent language translation algorithm[52]. In this algorithm, each node of the AST (ELM) is visited during a post-order tree traversal and is translated based on a set of rules. To support the logic necessary to execute the HF phenotype definition, we implemented rules for Boolean conjunctions (`AND`) and disjunctions (`OR`), temporal logic to calculate patient age, numeric comparison, the `Count` aggregate function, filtering data by value sets, and correlated queries, which

express relationships between data elements. Data model translation is performed during the creation of Circe criteria from ELM `Query` constructs.

3.3.3 Validation

The CQL on OMOP tool was validated in two ways - *cross-institutional* and *cross-platform.* First, the cross-institutional validation checked the accuracy of the translated phenotype logic when executed on two instances of the same CDM (OMOP). This was done at Northwestern Medicine (NM) and Weill Cornell Medicine (WCM), and we verified the results manually to evaluate if the phenotype logic was correctly applied (figure 3.2 pipeline 5). Second, we conducted a cross-platform validation to evaluate that consistent results were returned when the same phenotype logic was applied to the same synthetic dataset in two different execution pipelines - an OMOP environment, and an independent, FHIR-native CQL execution pipeline (figure 3.2 pipeline 6).

3.3.3.1 Cross-Institutional

The NM OMOP instance used for cross-institutional validation is a subset of the patient population at NM, specifically, those consented for the eMERGE network, and is generated from the NM EpicCare® EHR. The WCM OMOP instance includes the patient population at WCM and its affiliate NewYork-Presbyterian (NYP) hospital with at least one recorded visit, condition, and procedure. In the outpatient setting, WCM physicians use the EpicCare® EHR, and NYP uses the Allscripts Sunrise Clinical Manager for inpatient care.

Each institution selected a random set of 25 patients that were identified by CQL on OMOP as meeting the criteria of the HF phenotype (cases). Additionally, each institution selected 25 random patients who a) were not included as a HF case, b) had at least one

echocardiogram procedure, and c) had at least one relevant diagnosis code (non-cases). In review of the HF phenotype definition, we believed the highest chance of error in the translation of the logic was in the portion aligning diagnoses with encounters (**C3** and **C4** in figure 3.1). As this is a technical verification and not a clinical validation, we believed this would be more likely to identify implementation errors than a random selection of patients not meeting the case definition, as the majority of patients would simply be lacking diagnoses (given the overall expected low prevalence of HF). Cases and non-cases were selected from the respective OMOP databases using SQL scripts that were prepared collaboratively by the reviewers ahead of time.

At NM, one reviewer (LVR) evaluated the set of 50 cases and non-cases in OMOP. A random subset of 10 patient records (5 cases and 5 non-cases) was reviewed by a second reviewer (JAP). At NM, each reviewer used the ReviewR tool, which provides a graphical interface and filtering capabilities against an OMOP database[118]. At WCM, a similar process was followed with a primary reviewer (ETS) reviewing all 50 patient records and a second reviewer (PA) reviewing a random subset of 10 patient records. WCM reviewers accessed the OMOP database via SQL queries to retrieve the data elements needed for the results verification. Both institutions used the same SQL code to generate the random list of patients for review and followed the same written protocol. This code and documentation are available in the project's GitHub repository. We calculated Cohen's kappa[119] to measure inter-rater reliability, and determined overall system performance using precision and recall.

3.3.3.2 Cross-Platform

To assess cross-platform performance, we compared CQL on OMOP to the reference implementation of the CQL engine provided by the language authors[120], running against a HAPI FHIR server[121]. We used data for 1000 patients from the Centers for Medicare & Medicaid Services' (CMS) Data Entrepreneurs' Synthetic Public Use File (SynPUF 1k)[122]. Although synthetic, the dataset is intended to be representative of a typical claims dataset collected by CMS. The dataset was transformed into the OMOP CDM schema using the extract transform load (ETL) tool provided by the OHDSI community[123], and transformed into the FHIR format using the OMOP on FHIR tool[124].

We ran the HF phenotype definition using CQL on OMOP against an OHDSI instance containing the SynPUF 1k dataset and generated the resulting cohort of patients. We then ran the same HF CQL using the CQL reference implementation against a FHIR server containing the same SynPUF 1k data and compared the resulting patient cohorts. Performance (agreement between the two systems) was measured using Cohen's kappa.

3.4 Results

3.4.1 Cross-Institutional

The NM OMOP instance contained $8,657$ patients, of which 668 patients (7.7%) were identified by the HF phenotype definition, from which 25 were randomly selected for review. Of the 7,989 patients not qualifying for the HF cohort, 139 patients had at least one HF diagnosis and at least one echocardiogram, from which 25 were randomly selected as the non-case review cohort. Inter-rater agreement was $\kappa = 1.0$ between the two reviewers, and the CQL

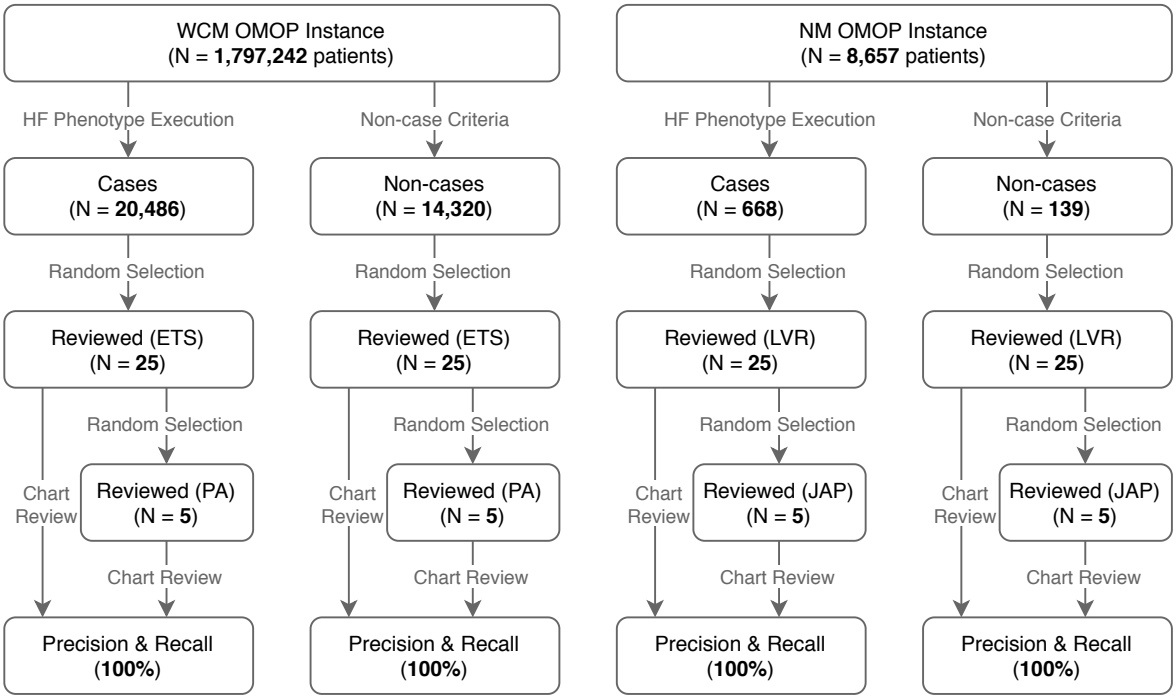


Figure 3.3: Results and validation flowchart for translation execution pipeline.

to OMOP translation execution pipeline achieved both precision and recall of 100%.

Of the approximately $1,797,242$ patients in the WCM OMOP instance, $20,486$ (1.4%) were in the HF cohort. There were 14,320 patients that matched our non-case criteria. The 25 cases and 25 non-cases randomly selected for review demonstrated precision and recall of 100%. Inter-rater agreement was again $\kappa = 1.0$.

3.4.2 Cross-Platform

After performing ETL on the SynPUF 1k dataset, the resulting OHDSI instance contained $147,186$ conditions, $55,261$ visits, and $137,522$ procedures for the $1,000$ synthetic patients. We confirmed the same counts of each data element after application of the OMOP on FHIR data transformation tool to verify no data were lost.

Running CQL on OMOP against an OMOP instance containing the SynPUF 1k dataset resulted in a cohort with 94 members (9.4%). Executing the same CQL using the CQL

reference implementation pointing to a HAPI FHIR server containing the same SynPUF 1k dataset represented as FHIR resources also generated a cohort containing 94 patients. Since patient IDs were kept consistent by the ETL and OMOP on FHIR processes, we were able to confirm that these cohorts were in complete agreement with $\kappa = 1.0$.

3.5 Discussion

We were able to express a HF phenotype algorithm as a CQL library, and demonstrate consistent execution across multiple institutions with different populations (NM and WCM), as well as different data platforms (OMOP and FHIR) representing the same synthetic patient population. Manual translation of the query logic was not required in this process, thereby, limiting the potential for error. Thus, CQL reduces duplication of effort, increases transparency and phenotype portability, and reduces variability. Furthermore, our selection of a clinically-purposed language (CQL) will facilitate extension of this approach beyond research phenotypes to clinical and analytical needs of a LHS.

CQL libraries modularize logic using named statements and functions, facilitating reuse, which is highly beneficial for phenotyping as it enables defining cases and controls which often have shared logic. Well-constructed libraries can then also extend past binary case/control classification to include *suspected* cases, sub-phenotypes, related phenotypes, and even groups of phenotypes. Libraries can also be parameterized, which can be used to support local customization, within well defined bounds, to match site-specific clinical and operational procedures.

Although our translation of the HF phenotype logic performed with high precision and

recall, we note differences between the conceptual models used by CQL and the OHDSI Circe library. In Circe, phenotypes have specific entry and exit events, and the concept of *observation period* is used to determine cohort membership, which are not explicit concepts in CQL. Circe and CQL also have different internal AST representations. CQL uses a traditional AST with very simple nodes, and a topology correlated with the complexity of the represented logic, while Circe uses nodes that encode additional information, and generally has a simpler topology, only using the tree to encode conjunction and disjunction, occurrence count restrictions, and temporal correlations. CQL's more traditional AST structure lends itself well to language applications like translation and interpretation, while the structure of Circe may simplify SQL query construction and make it easier to build user interfaces to author cohort definitions. Lastly, criteria in Circe can be manually grouped into *inclusion rules*, which supports the generation of attrition statistics and visualizations. Since this grouping requires human intervention, it is not possible to generate meaningful inclusion rules in CQL on OMOP without introducing further conventions (e.g., annotations), which we decided against to ensure cross-platform support for CQL-based phenotypes.

Using the FHIR data model for data element references resulted in several advantages. Due to the popularity of FHIR, a data model translation already existed for the OMOP CDM, which reduced the amount of implementation work necessary for CQL on OMOP. The HF phenotype logic references unambiguous data elements such as conditions and procedures, which are highly mature entities in the FHIR specification, and have very clear mappings to the OMOP CDM. However, some phenotype definitions may reference more nuanced data elements that may be more difficult to translate. While it may take more upfront work to deal with these issues in the CQL engine, this work will only have to be done once per target

platform, reducing overall work required for phenotyping, along with phenotype variability.

The reduced expressiveness of Circe, compared to CQL, limited our current CQL on OMOP implementation. As many institutions within the OHDSI community develop cohort definitions using SQL, R, or other languages as opposed to Circe, this may have been a purposeful limitation by the OHDSI developers. CQL approaches the expressiveness of a general-purpose programming language, and as such can express arbitrary arithmetic, and has many aggregate functions not supported by Circe (e.g., `Sum` and `PopulationStdDev`). To address these limitations Circe can either be extended to be more expressive, or CQL on OMOP could bypass Circe and the OHDSI API entirely and access the database directly. While the latter approach would enable the full expressivity of CQL, the former approach is more desirable, since it maintains compatibility with all of the phenotyping and other tools in the OHDSI community.

An important limitation of the CQL language itself is that it is optimized for rule-based logic using structured data elements, and does not explicitly define any mechanism for natural natural language processing (NLP) or integration with machine learning (ML) methods. Both of these techniques are important to the task of phenotyping[32,125], and being limited to structured data and deterministic algorithms is a significant restriction. However, CQL does provide a mechanism to integrate with external systems using an approach called *foreign function invocation* (FFI). FFI enables a given engine implementation to make functions available to the CQL library author that execute code in an arbitrary environment, such as an NLP or ML pipeline, and make the execution results available in CQL. Furthermore, CQL can leverage existing NLP systems that already utilize the FHIR standard to provide standardized models and normalization rules for integrating unstructured data[27]. These

features could be used to develop extremely high fidelity phenotypes that make use of the latest NLP and ML algorithms.

We acknowledge additional limitations within our work. First, our evaluation was performed using a single phenotype (heart failure), and does not include support for all operators within CQL at this time. We selected the HF phenotype definition given its use of multiple data elements (diagnoses, encounters, procedures, demographics), temporal logic, and aggregate functions (`Count`), which represents commonly used building blocks across other phenotype definitions. Second, we recognize that the validation of 50 cases and 50 non-cases may be seen as minimal, and that our selection of non-cases is not representative of all patients not identified by the HF algorithm as cases. Given that our focus was on a technical verification and not a clinical validation, we believe that our review allowed us to focus on the most probable sources of error. Third, the upfront cost of developing a CQL engine for a new target platform may be prohibitive, and potential implementers of the proposed approach would need to balance this cost against potential benefits. If the implementer has no desire to share or reuse existing phenotype definitions, or if cross-platform phenotyping is not a requirement, then using existing query tools may be more appropriate.

Despite the above limitations, we have shown that CQL can be used to represent and execute a clinically validated phenotype, using our CQL on OMOP engine. Due to its highly expressive nature, CQL could be used to represent longitudinal phenotypes with highly complex data relationships. Furthermore, in our experience, the CQL language specification (with its canonical AST) makes implementing language engines against arbitrary data platforms relatively easy. Therefore, CQL is a promising candidate as a formal phenotype representation standard that supports cross-platform execution.

3.6 Conclusions

The task of EHR-driven phenotyping is critical to biomedical knowledge generation, which supports the learning health system. Current techniques suffer from portability and scalability issues, requiring human intervention. This leads to errors, variability, lack of transparency, and greatly reduces potential throughput. To address these issues, we investigated CQL as a candidate language for representing clinical phenotype definitions, and demonstrated execution against multiple data platforms without local customization. We believe this approach could speed up phenotyping, regardless of the underlying data platform. Using a computable standard representation would also reduce duplication of work and potential for human error, and enable the large scale phenotyping needed for learning health systems.

In future iterations of the PhEMA project we plan to extend CQL language support in CQL on OMOP, translate additional clinical phenotypes into CQL, use CQL-based phenotype definitions in clinical research studies, and extend existing phenotype authoring tools to generate CQL. Furthermore, we plan to develop CQL execution engines against other data platforms, such as the Informatics for Integrating Biology and the Bedside (i2b2) platform[126], and extend CQL to support NLP and ML. We will continue this work with existing phenotyping communities to publish methods and tools with the ultimate goal of convergence on a unified system to support high-quality and high-throughput phenotyping efforts.

CHAPTER 4

PHEMA WORKBENCH: A PLATFORM-INDEPENDENT FHIR-NATIVE EHR-DRIVEN PHENOTYPING TOOLBOX

4.1 Introduction

Many studies designed to generate biomedical knowledge begin with cohort identification, also called electronic health record (EHR)-driven phenotyping, which is a resource-intensive process involving many stakeholders that must often be repeated every time the same cohort is studied[8,23]. Given increased focus from the research community in the past decade, many different approaches and tools have been developed that attempt to reduce duplication of work, decrease the time investment required, and limit the potential for human error[18,126–134]. These approaches and tools have grown in response to specific needs, which by and large has resulted in a fragmented ecosystem of tools that are not interoperable. For more comprehensive suites of tools, they have been developed to work within a single data model, which can hinder their adoption in other contexts. In this work, we present a standards-based representation for cohort definitions (also called phenotype definitions) and an accompanying tool that together facilitate interoperability in the existing heterogeneous EHR-driven phenotyping environment. We hypothesize that the proposed methods will reduce duplication of work and thereby increase the velocity of biomedical knowledge generation.

4.2 Background

4.2.1 Computable Phenotyping

EHR-driven phenotyping is conceptually a three-step process (although the steps may be conducted iteratively). The first step is defining the phenotype definition, or just *phenotype* (called *authoring*); the second step is executing the phenotype against some data repository (called *execution*); the third step is to evaluate or validate the correctness of the resulting cohort (called *validation*). Authoring is usually a collaboration between clinical experts and informaticists to elucidate requirements specific enough to proceed to the execution step. Informaticists perform the execution step, sometimes in collaboration with database analysts, in order to extract the cohort from the data source by using, for example, SQL or programming code custom-built for the specific data source. Validation is typically done by clinical experts or trained chart abstractors and involves reviewing the entire medical record. Although some recent advances have explored how to identify relevant phenotype logic using automated methods[135,136], the authoring step cannot be fully automated. This is because it requires input from clinical domain experts in order to establish a clinically correct definition. Likewise, the third step requires clinical expertise to confirm that the resulting cohort members match the clinical criteria. However, the second step (execution) is repeated every time the cohort of interest is used in biomedical knowledge generation, and is the step that can be partially or fully automated.

4.2.2 Current Strategies

The EHR-driven phenotyping research community in general, and the Phenotype Execution Modeling Architecture (PhEMA)* project in particular, has identified many requirements for the use of computable artifacts to automate the process of EHR-driven phenotyping[16,34,96,137,138]. These requirements include using structured and standardized data representations, using human-readable and computable representations for cohort criteria, providing interfaces for external software, and maintaining backwards compatibility[16]. If the artifacts generated by the authoring step meet these requirements, this would allow the execution step to be partially or fully automated. This will increase the velocity at which clinical research can be conducted and reduce the risk of human error introduced in the process of translating and running phenotypes.

At least two strategies have been employed to meet these requirements, namely, the use of common data models (CDMs) and the use of generic formal logic representation and execution environments. The use of CDMs involves extracting data from existing data repositories, transforming it to meet the requirements of the CDM, and then loading it into the CDM database. In some cases, it may also involve mapping terminologies and coded data elements into the standardized vocabularies required by the CDM. Once data is in the CDM, any computable phenotyping artifacts created at one site, such as SQL or programming code, can be automatically used at any other site using the same CDM[28]. This approach has been successfully used by the Observational Health Data Sciences and Informatics (OHDSI) program[18], the electronic Medical Records and Genomics (eMERGE) Network[12–14], the National Patient-Centered Clinical Research Network (PCORnet)[31], the Accrual to Clinical

*https://projectphema.org

Trials (ACT) Network[102], and others[28,37]. One disadvantage of this approach is the initial time investment involved in preparing the data, which may mean abandoning or recreating any artifacts developed for the old data format. It might be possible to maintain both the old and new data formats, but this comes with maintenance and operational costs. Additionally, phenotype definitions created for one CDM cannot be used to generate cohorts in another CDM. The use of generic logic execution environments such as KNIME and Drools have been demonstrated to be successful but are not based on any healthcare standard and may also require a preparatory data transformation step[16,25,104]. Formal representations such as the HL7 Health Quality Measure Format (HQMF), the Clinical Decision Support Knowledge Artifact Specification (CDS KAS), and Arden Syntax have also been used successfully for representing clinical logic[25,139,140], but only Arden Syntax has a natively human-readable representation, and none have a convenient user interface, making them difficult to implement without the development of custom tools.

4.2.3 Existing Tools

There are several existing tools for authoring and executing computable phenotype and cohort definitions. Some tools are based on the CDM strategy, while others are data model independent, but require data elements to be formally mapped or defined in terms of the local data model before the tool can be used. One highly mature and widely used tool in the clinical research informatics community is the Atlas tool provided by the OHDSI program*. This tool is built on the Observational Medical Outcomes Partnership (OMOP) CDM[18], and provides users with an interface allowing them to specify logical criteria in a

*https://github.com/OHDSI/Atlas

point and click manner. Clinical data elements can be selected and filtered, and criteria can be correlated using Boolean and a limited set of temporal and aggregate operators. Additionally, cohort definitions can be exported in JSON format and shared with other OMOP users, who can import the definitions and identify a cohort meeting the same criteria. However, this format is not a formally defined standard, and cannot, for example, contain sub-phenotype definitions. Another popular CDM and suite of tools, Integrating Biology and the Bedside (i2b2)[126], provides a drag and drop interface for cohort identification, but does not use a formal standard or have the ability to export or import these definitions natively. A nascent phenotyping method, called Phenoflow, supports the development of portable phenotypes using a formal representation[141]. While the overall workflow process described in a Phenoflow phenotype definition is based on a standard (the Common Workflow Language (CWL)[142]), the individual steps are implemented using custom programming code, which is not standards-based.

4.2.4 Phenotype Repositories

The OHDSI community maintains a repository of cohort definitions called the Phenotype Library*, which can be imported into any OHDSI instance and executed without requiring local customization. CALIBER is a phenotype definition library in the UK that contains over 350 phenotype definitions, which are provided as structured value sets and narrative descriptions of phenotype logic[143]. Also in the UK, the Phenoflow Phenotype Library† has over 330 phenotype definitions consisting of directly computable artifacts, namely CWL and custom programming code. The Phenotype KnowledgeBase (PheKB), associated with the

*https://data.ohdsi.org/PhenotypeLibrary
†https://kclhi.org/phenoflow/phenotype/all/

eMERGE network, is the most extensive and most mature phenotype definition library in the United States, with 77 publicly available phenotype definitions[32]. Like CALIBER, the phenotype definitions in PheKB do not provide computable artifacts, except for a handful of exceptions, which provide computable artifacts that do not conform to any healthcare standard, such as custom programming code, or KNIME[94] workflows.

4.2.5 PhEMA Approach

In this work we present an approach that combines elements of the CDM approach with elements of the generic logic approach, taking advantage of the benefits of both and mitigating some of the disadvantages. Our approach does not require additional data preparation, works across data platforms, uses established healthcare standards, and uses both human-readable and computable phenotype representations. We demonstrate and evaluate a tool, called the *PhEMA Workbench*, that allows users to author phenotype logic, assemble value sets by integrating with existing tools, and execute phenotypes against existing data stores without requiring manual translation. An essential goal of this approach is to enable incremental adoption and interoperability with existing systems.

4.3 Methods

4.3.1 Formative Research

Requirements for the Workbench were elucidated using two separate user research studies. The first version of the PhEMA Phenotype Authoring Tool (PhAT) was developed using participatory design with fifteen end users. The requirements included that both textual and visual representations of the phenotype definition should be available, details of the execution

process should be provided, a flexible logic expression language should be used, and a library of standard phenotype definitions should be available. Additionally, common terminologies and value sets should be available from within the tool[144]. For the tool presented here, we conducted user research by showing wireframes to five expert users and conducting semi-structured interviews. We again identified that a graphical view is desirable, and that value sets and a library of phenotype definitions should be available from within the tool.

4.3.2 Standards-Based Representation

We propose a fully Fast Healthcare Interoperability Resources (FHIR)-native representation for phenotype definitions. The representation uses FHIR `ValueSet` and container resources (`Bundle` and `Composition`), and the Clinical Quality Language (CQL), which is one of the formal logical expression languages referenced in the FHIR specification. Our representation could be adopted by any system, as it includes no proprietary technology. Inclusion and exclusion logic is expressed using CQL, which produces an unambiguous and human-readable representation. CQL source files are contained in `Library` resources as defined in the FHIR Clinical Reasoning Module. Each phenotype has one main CQL library containing the "Case" definition, and any number of helper libraries. Lists of codes from standard terminologies are represented using `ValueSet` resources as defined in the FHIR Terminology Module. A `Composition` resource is used to collect all `Library` and `ValueSet` resources into a single document that contains all the artifacts necessary to describe the phenotype definition fully. The `Composition` can optionally reference additional metadata, such as an `Organization` resource representing the phenotype author, a `Basic` resource containing all metadata available in PheKB, or other relevant artifacts. By convention, the `Composition`

section entry for the main phenotype `Library` is titled "Phenotype Entry Point" to indicate to the executing system where to find the "Case" definition. As defined in the FHIR Foundation Module, the `Composition` and all other resources are contained within a `Bundle` resource used to persist or transmit the fully-specified computable phenotype definition. This representation enables FHIR operations to be used for storage, retrieval, and execution. For example, the `$cql` operation defined in the Clinical Practice Guidelines (CPG) implementation guide can be used to execute the phenotype, and the `$document` operation defined in the FHIR Foundation Module can be used to assemble the complete phenotype definition based on the `Composition` resource.

4.3.3 System Description

4.3.3.1 Architecture

The PhEMA Workbench is designed as a standalone tool that can be used for phenotype authoring, execution, and publishing to a shared phenotype definition repository. The architecture (figure 4.1) is designed to function without requiring any changes to existing phenotyping tools or infrastructure, integrates with OHDSI out of the box, and supports popular and emerging standards, namely FHIR and CQL. The components include a web application that runs in the user's browser, a backend API to support integrating with existing systems, as well as services for phenotype development and testing. The application is written using TypeScript, a strongly typed language developed by Microsoft that compiles to JavaScript, and the API is written in Java. All code is open source and available online in the PhEMA GitHub organization*. The tool used for testing during authoring is CQF

*https://github.com/PheMA

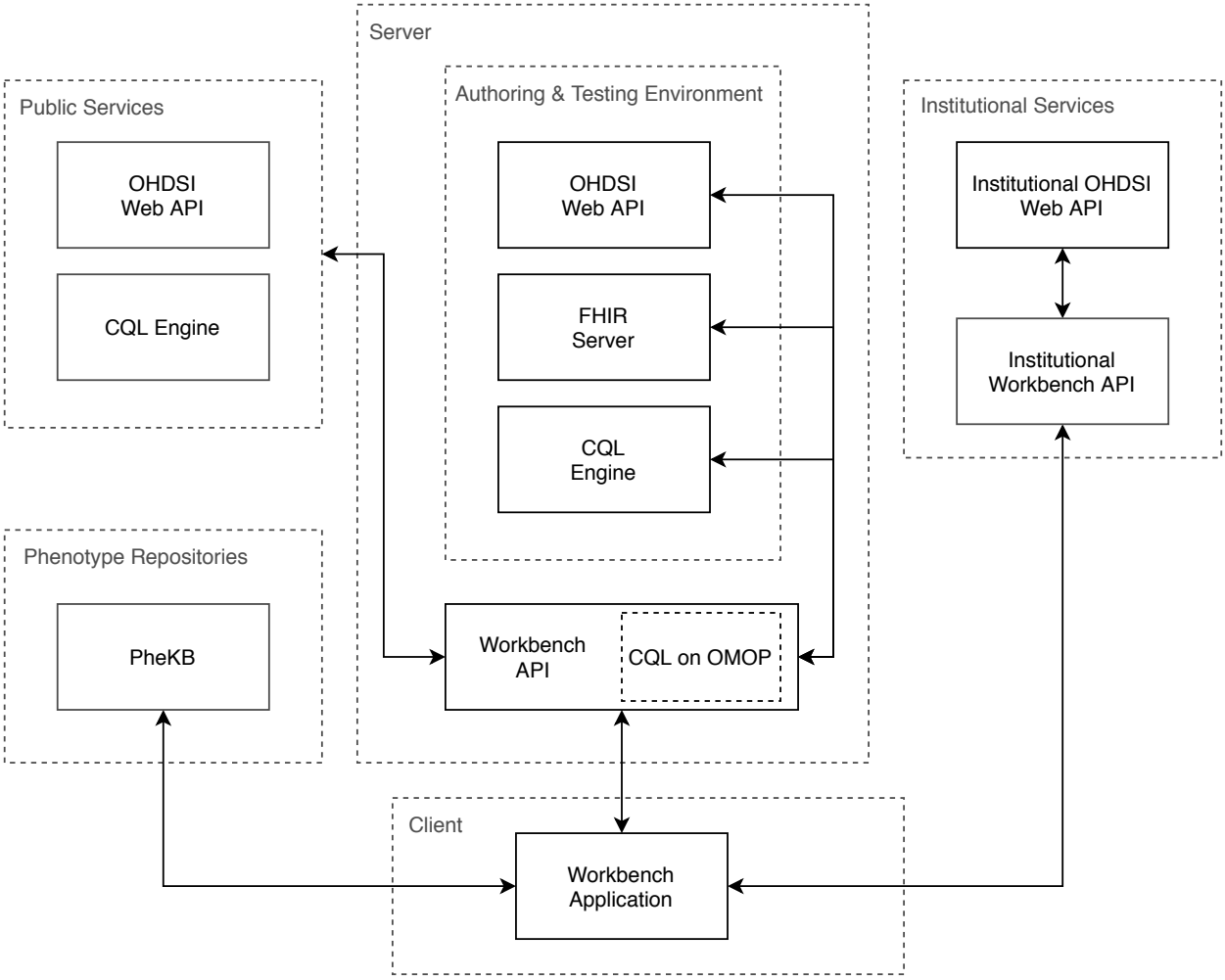


Figure 4.1: System architecture. Services in the box labeled *Server* run on the PhEMA server, and are accessible via the public internet. The box labeled *Client* runs in the browser on the user's machine, and is accessed by navigating to a specific URL on the PhEMA server. Optional publicly accessible third party services such as additional FHIR servers or OHDSI Web API instances are shown in the box labeled *Public Services*. The *Phenotype Repositories* box shows repository services (currently only PheKB). The *Institutional Services* box shows services that run behind institutional firewalls.

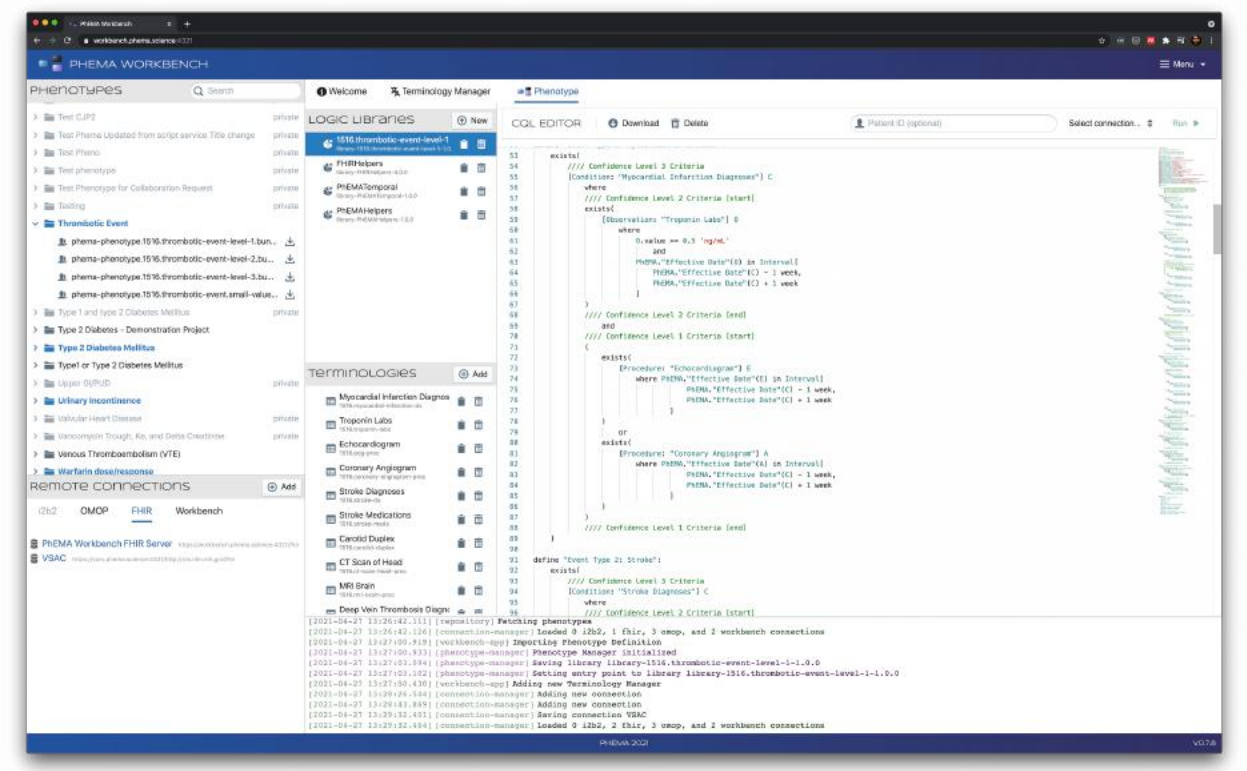


Figure 4.2: CQL Editor.

Ruler*. It consists of a HAPI FHIR server[121], and an implementation of the FHIR Clinical Reasoning Module and CPG implementation guide, which both make use of the reference implementation of the CQL engine[120].

4.3.3.2 Features

The first essential Workbench component is the CQL editor (figure 4.2), which runs on the client and allows users to write CQL expressions. The CQL editor provides syntax highlighting and supports executing CQL against any environment capable of CQL execution, including the provided testing environment described below.

The application also provides a terminology manager component (figure 4.3), which allows

*https://github.com/DBCG/cqf-ruler

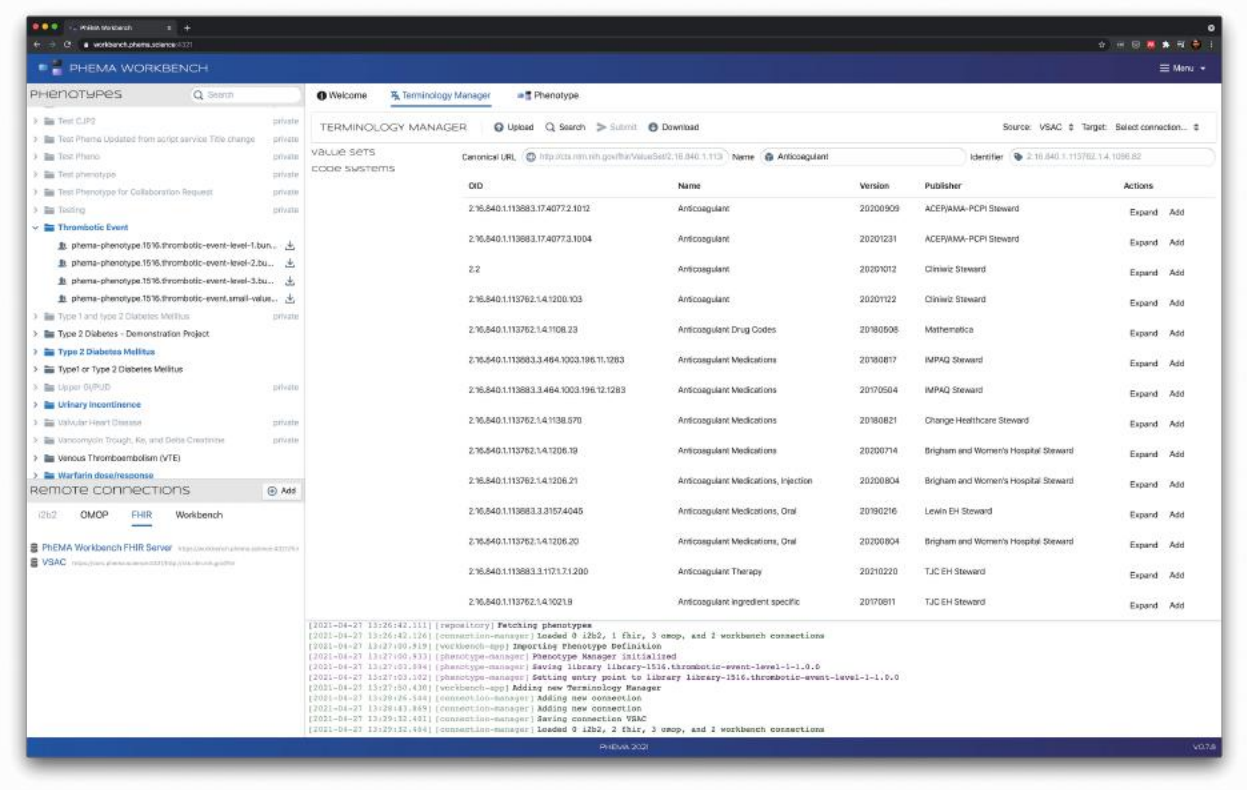


Figure 4.3: Terminology manager.

a user to assemble a collection of value sets into a FHIR `Bundle` of `ValueSet` resources. Value sets can be uploaded by dragging and dropping them into the application, or by selecting a set of files on the filesystem. The supported formats include `ValueSet` or `Bundle` resources, concept sets exported from the OHDSI Atlas interface (either the full ZIP file or individual CSV files), or a custom PhEMA CSV format. The user is also able to directly search the National Library of Medicine's Value Set Authority Center (VSAC) FHIR server[47], expand the results to inspect the codes in the value set, and add one or more of the search results into the `Bundle` if appropriate.

It is helpful for an author to test their CQL against test data during phenotype development to ensure that the expressed logic performs as expected. The Workbench environment

supports this by providing a FHIR server that can be loaded with synthetic data to be used for testing. During the development of the phenotype used in this study, we created 21 different test patients, each with associated data to test the various criteria in the phenotype definition. This process, sometimes referred to as test-driven development (TDD), gives the author confidence that the developed phenotype definition is semantically correct. The CQL is executed on the FHIR server using an extended operation called `$cql`, provided by CQF Ruler.

The Workbench integrates with the existing PheKB phenotype repository (top left panel in figures 4.2–4.4). The integration supports listing all publicly available phenotypes, importing phenotype definitions that are represented using the proposed FHIR-native standard, and publishing new phenotype definitions. This integration is accomplished by using the API provided by the PheKB application.

Additionally, the Workbench currently supports OHDSI and FHIR as execution targets. At execution time, the Workbench API processes the complete FHIR-native phenotype, translates it to the appropriate representation using CQL on OMOP[37] in the case of the OHDSI target, and executes the logic against the target data store (figure 4.4). These execution targets evaluate the complete phenotype definition against a data store, and establish the corresponding cohort. The Workbench also supports simple CQL execution, in which an individual CQL library is evaluated outside of the context of a phenotype. Additionally, it is possible to generate an OMOP compliant SQL script representing the phenotype definition. This supports the use case in which a research site uses the OMOP database, but does not use the OHDSI Web API.

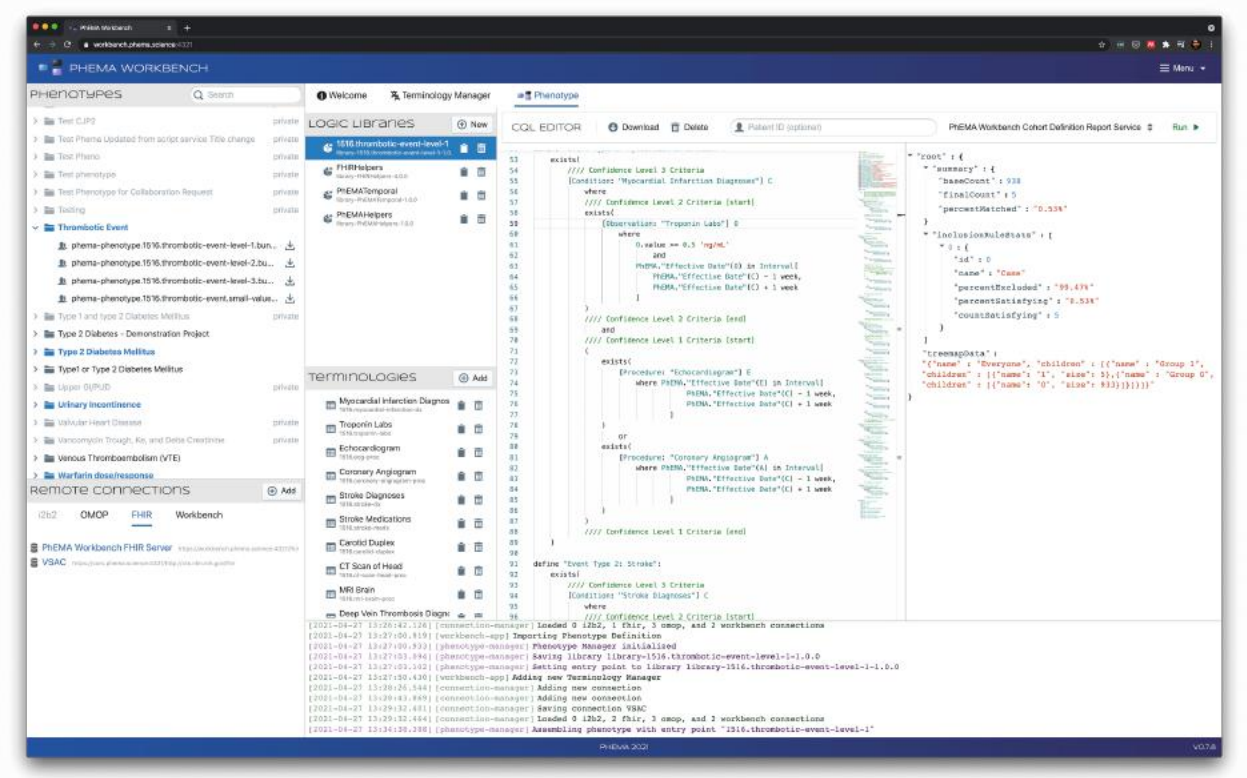


Figure 4.4: Automated execution. Results shown in the right-most panel.

4.3.4 Experimental Setup

4.3.4.1 Phenotype Definition

To test the PhEMA Workbench, we selected a thrombotic event (TE) phenotype developed by clinicians at Weill Cornell Medicine (WCM). The phenotype identifies patients that have experienced one or more of ten different thrombotic events, such as myocardial infarction, stroke, pulmonary embolism, and others. The definition for each event has three different criteria sets that correspond to different confidence levels. The lowest confidence level (level 3) only requires that a patient has one of a given set of International Classification of Diseases versions 9 (ICD-9) or 10 (ICD-10) codes. For example, for the placenta thrombosis event, a patient meets the level 3 criteria if they have an ICD-9 code beginning with 663.6 or the

ICD-10 code O43.813. To meet the level 2 criteria, patients need to have a specific drug or lab order, depending on the specific thrombotic event, in addition to an ICD code. For the highest confidence level (level 1), patients must meet the level 2 requirements, and also have an order for a specified procedure, and in some cases an additional lab or drug order. An additional requirement for each event type is that all criteria must be effective within a one-week period. The phenotype definition was shared with the PhEMA collaborators in the form of a textual narrative description. Table 4.1 lists all criteria for the individual events exactly as supplied in the narrative description.

Event Type	Confidence Level 3	Confidence Level 2	Confidence Level 1
MYOCARDIAL INFARCTION	ICD-9 code of 410.X or ICD-10 code of I21.X anywhere in the patient's EHR	CL 3 + troponin of 0.5 or higher	CL 2 + echocardiogram, ECG, or coronary angiogram
STROKE	ICD-9 code of 434.11 or ICD-10 code like I63.[0-3]% or ICD-10 code like I63.[5-9]% anywhere in the patient's EHR	CL 3 + order for aspirin or clopidogrel + carotid duplex order + echocardiogram order	CL 2 + neurology consult order + CT head or MRI brain order
DVT	ICD-9 code of 453.4X or ICD-10 code of I82.4X OR any instance of a DVT sentinel phrase in any note	CL 3 + D-Dimer fibrin lab result OR order for anticoagulant	CL 2 + LE duplex report with positive DVT sentinel phrase
PE	ICD-9 code of 415.1X or ICD-10 code of I26.9X anywhere in the patient's EHR	CL 3 + new anticoagulant prescription + LE duplex report WITHOUT positive DVT sentinel phrase	CL 2 + CT chest or VQ scan order
MESENTERIC-SPLANCHNIC THROMBOSIS	ICD-9 code of 557.0X or 444.89 or ICD-10 code of I81.9X/K55.0X/I82.0X/I74.8X anywhere in the patient's EHR	CL 3 + new anticoagulant prescription	CL 2 + Sonogram order OR CT chest order OR CT abdomen order OR MRI order + D-Dimer fibrin lab result + New anticoagulant order
SUPERFICIAL VEIN THROMBOSIS		ICD-9 code of 453.6X or 451.89 or 453.82 or ICD-10 code of I82.81 or I80.0X or I82.61	CL 2 + new order for anticoagulant
OTHER ARTERIAL THROMBOSIS		ICD-9 code of 444.1X or 444.22 or 453.3X or ICD-10 code of I74.X or I65.1X or I82.3X	CL 2 + new order for anticoagulant
PLACENTA THROMBOSIS		ICD-9 code of 663.6X or ICD-10 code of O43.813	CL 2 + new order for anticoagulant
CENTRAL NERVOUS SYSTEM (CNS) THROMBOSIS		ICD-9 code of 437.6X or ICD-10 code of I67.6X	CL 2 + new order for anticoagulant
ENDOCARDIAL THROMBOSIS		ICD-9 code of 996.71 or 444.9X or ICD-10 code of I34.8X or I51.3X	CL 2 + new order for anticoagulant

Table 4.1: Thrombotic event phenotype criteria.

CL – Confidence Level

4.3.4.2 Authoring

We used the experimental architecture illustrated in figure 4.5. In the first step, authoring and publishing was done by a single author (PSB). This was done by manual interpretation of the TE phenotype narrative description, and writing the corresponding CQL statements using the Workbench CQL editor. We omitted criteria that required natural language processing (NLP), as this data is not available to OMOP cohort definitions. Test data was created using the CQL Testing Framework tool created by the Agency for Healthcare Research and Quality (AHRQ)*, which allows users to generate FHIR resources using a light-weight YAML file in which many fields are inferred using sensible conventions. Value sets were generated in FHIR format using a custom script that expands a CSV file with extensional and basic intensional (e.g., regex) definitions into a set of complete FHIR `ValueSet` resources. The value set for anticoagulant drugs was imported directly from the VSAC FHIR server. All value sets were assembled into the final phenotype `Bundle` using the Workbench terminology manager component. Once all CQL logic was written and tested, and appropriate value sets were assembled, the phenotype was packaged using the proposed FHIR-native representation and published to PheKB.

*https://github.com/AHRQ-CDS/CQL-Testing-Framework

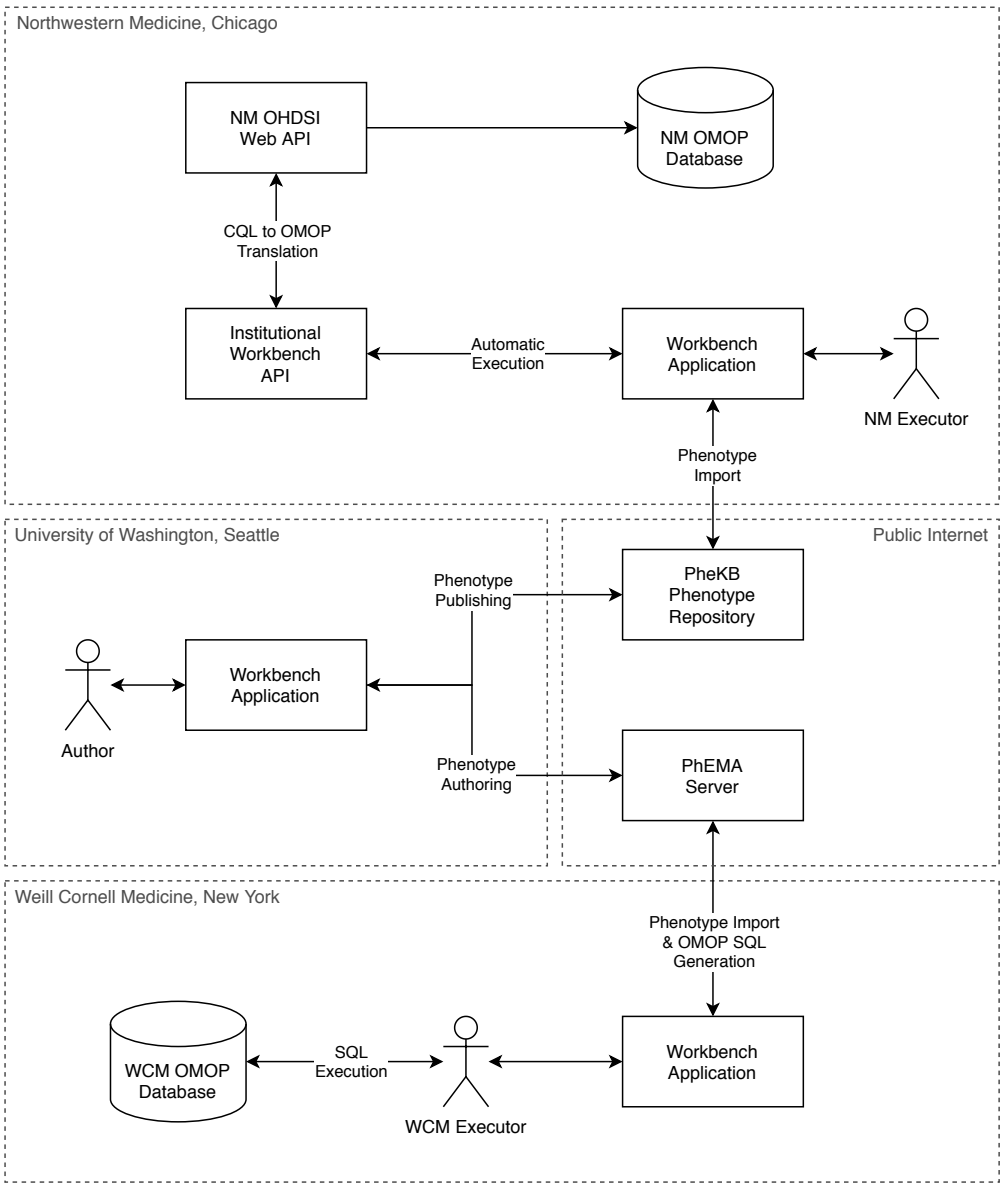


Figure 4.5: Experimental architecture. The phenotype was authored by PSB at the University of Washington and published to the PheKB repository in the proposed FHIR-native format using the Workbench application. LVR used the Workbench at Northwestern Medicine (NM) to execute the phenotype automatically using an institutional instance of the Workbench API, since the NM OHDSI Web API is not accessible from the PhEMA Server. At Weill Cornell Medicine (WCM), PA used the Workbench application to generate an SQL script and executed it against the WCM OMOP database manually.

4.3.4.3 Execution

In the second step, users at two different institutions, LVR at Northwestern Medicine (NM), and PA at WCM, imported the phenotype from PheKB into the PhEMA Workbench application. At NM, the phenotype was executed directly against the NM instance of the OHDSI Web API from the Workbench application, without any local customization. An institutional instance of the Workbench API was used to bypass NM firewall restrictions. At WCM, there is no instance of the OHDSI Web API running, so instead of directly executing the phenotype, an SQL script was generated by PA using the SQL execution target available in the Workbench application. This SQL script was then manually executed again the WCM OMOP database.

4.3.4.4 Validation

The OMOP database at NM contains data from a subset of the patient population that have consented to participate in the eMERGE Network. The data is sourced from the NM EpicCare® EHR system. The WCM OMOP database contains data for patients from both WCM and NewYork-Presbyterian (NYP) hospital that have at least one visit, condition, and procedure recorded. The EHRs used at WCM and NYP are EpicCare® and Allscripts respectively. We considered cases to be only those patients matching the confidence level 1 criteria. We used patients matching the confidence level 2 criteria (but not matching confidence level 1) as non-cases instead of selecting non-cases completely at random. This is because we believe that failures are most likely to occur at logical edge cases, so we wanted to validate these situations in particular. This validation protocol was shared in advance, along with an SQL script to randomly select cases and non-cases, and a data entry form for

reviewers to capture which criteria were met by each cohort member (if any).

At NM, a manual review of 25 cases and 25 non-cases was performed by LVR, who manually determined whether or not the patient had the appropriate data to meet the confidence level 1 criteria for at least one of the thrombotic event types. A second author (JAP) conducted a confirmatory review of 5 cases and 5 non-cases, and was blinded to the results of the first reviewer. At WCM, two primary reviewers (ETS and SA) both reviewed and manually verified the same set of 25 cases and 25 non-cases, with a secondary reviewer (PA) resolving any discordant determinations. We report precision, recall, and inter-rater agreement using Cohen's kappa[119] as performance measures for each site.

4.4 Results

4.4.1 Phenotype Definition

The resulting thrombotic event phenotype logic consisted of 11 CQL statements, one for each thrombotic event type, and one for disjunction of the other 10. The phenotype definition had 24 value sets containing a total of 834 codes for the various coded data elements referenced by the phenotype logic. Four different data sources were used, namely lab values, procedure orders, diagnoses, and drug orders, represented by the `Observation`, `Procedure`, `Condition`, and `MedicationRequest` FHIR resources respectively. During the authoring process, 21 test cases were created to test the phenotype definition logic. The resulting phenotype definition, including criteria logic and value sets, as well as the test cases and data, are available in the project repository on GitHub*.

*https://github.com/PheMA/thrombotic-event-phenotype

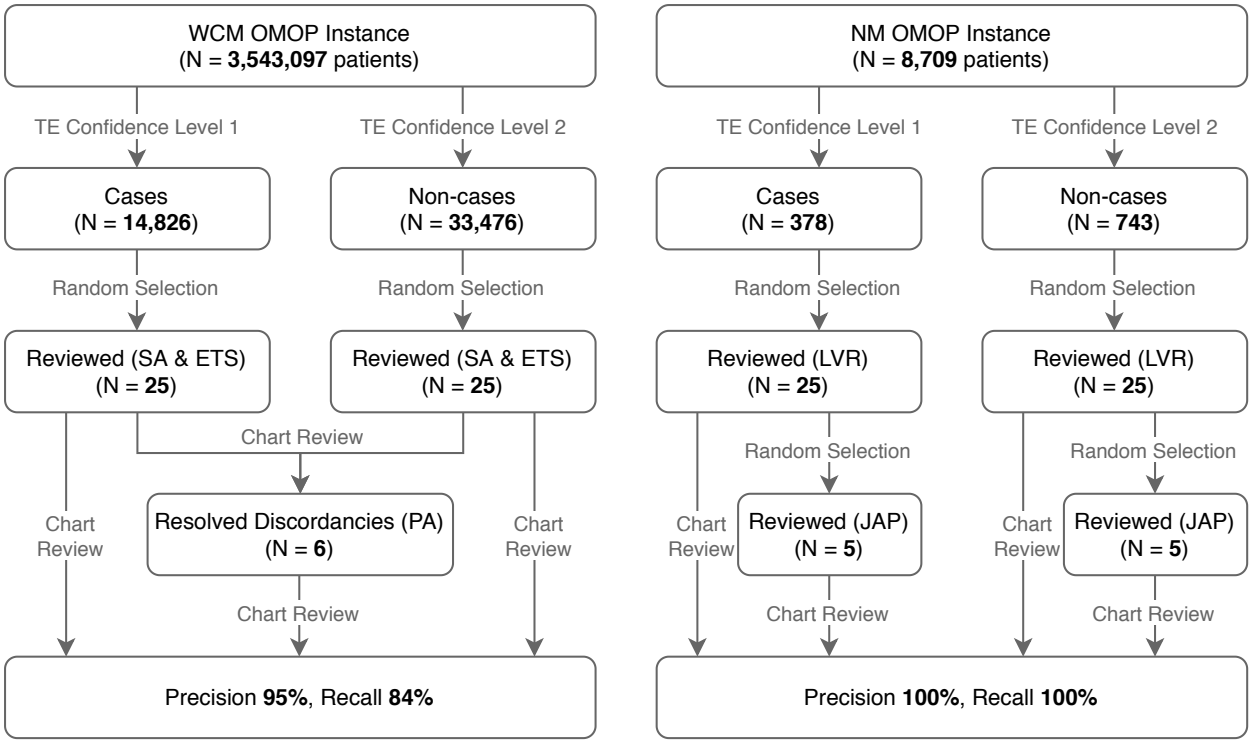


Figure 4.6: Results of the manual review process.

4.4.2 Validation

A summary of the validation process is shown in figure 4.6.

4.4.2.1 Northwestern Medicine

There were 8,709 total patients in the NM OMOP database, and 378 (4.34%) were identified by the FHIR-native phenotype to meet the TE confidence level 1 criteria, and 743 (8.54%) were identified to match the confidence level 2 criteria. Of the 25 cases and non-cases randomly selected for review, all were determined by the first reviewer to have been correctly identified. The secondary reviewer confirmed this through a review of 5 random cases and non-cases. The two reviewers were thus fully concordant, and manual review confirmed that all patients were correctly identified by the TE phenotype definition. Precision and recall

are therefore both 100%, and $\kappa = 1.0$

4.4.2.2 Weill Cornell Medicine

The WCM OMOP database contained a total of 3,543,097 patients, of which 14,826 (0.42%) were identified as cases and 33,476 (0.94%) as non-cases using the SQL script generated by the PhEMA Workbench. The two primary reviewers who reviewed all 25 cases and non-cases were discordant in 6 instances (15%), resulting in $\kappa = 0.74$. These discrepancies were resolved by the secondary reviewer, and of the 25 cases, 22 (88%) were manually confirmed to match the TE confidence level 1 criteria, and of the 25 non-cases, 24 (96%) were confirmed as not matching the confidence level 1 criteria. This results in a precision of 95% and a recall of 84%.

4.5 Discussion

We have demonstrated that a FHIR-native representation and platform-independent tool can be used for computable EHR-driven phenotyping, and achieve results comparable to other methods. We determined that the reason for the misclassified cohort members at WCM was due to how the OHDSI Web API performs concept searches, which occur during the CQL to OMOP translation step. Concepts are searched based on prefix matches, which in a few cases can return unrelated concepts with shared prefixes. This issue will be resolved in the next version of the CQL on OMOP tool, but even with this problem, a precision of 95% is still achieved. In addition, the PhEMA Workbench can integrate into existing clinical informatics research infrastructure without requiring any changes to currently used tools, and can in fact complement them. Furthermore, since the process is partially (in the case

of WCM) or completely (in the case of NM) automated, the time investment required is drastically reduced, and potential for human error is essentially eliminated.

Every step of the phenotype authoring process was done using only FHIR and CQL, which are open standards developed by Health Level Seven International (HL7). These standards are widely adopted, with the US recently adopting legislation that mandates the use of FHIR for healthcare data exchange, and CMS using CQL to represent the clinical quality measures required for reimbursement. Many EHR vendors and other systems support these standards, with adoption likely to increase. As such, our use of a FHIR-native representation is expected to remain compatible with existing and new systems over time. Furthermore, while CQL is data model independent, our choice to use FHIR for representing clinical data elements has additional advantages. First, much work has been done to map the FHIR data model to other widely used data models, such OMOP, i2b2, and others[46], which means that our phenotypes can be easily translated to those data models using standardized mappings.

CQL as a logical expression language is highly expressive, and we have demonstrated in this work and elsewhere[37] that it is capable of representing clinically validated phenotype definitions. Since CQL is a formal language, it also eliminates ambiguity that may result in variability of implementations. The language is designed specifically for the clinical domain, and thus has functionality tailored for representing clinical logic, such as the full set of temporal operators defined by Allen's interval algebra[145], aggregate operators common for quality measures and decision support, and uncertainty semantics to deal with missing data. Additionally, CQL provides integration points that can be used to integrate with external systems such as NLP pipelines, machine learning models, or other third party services.

Use of standards enables both technical and conceptual decoupling of concerns in the

phenotyping ecosystem, which is a widely used strategy in the technology industry to increase scalability, reliability, and extensibility[96,146]. For example, CQL is data model independent, meaning that the language engine is focused on logic execution, and delegates to a data provider module to collect the relevant data. This means that additional modules and execution targets can be developed for different data sources without requiring logic to be rewritten. Additional specialized tools can be developed for individual tasks, such as value set creation or visualization. Different CQL libraries can be assembled in a modular way to define a phenotype, which is an approach that allows for code reuse, and for easy localization, by having site-specific logic contained in a library that can be used as a "drop in" replacement for more general libraries.

Using a phenotype representation based on published standards enables the decoupling of phenotype authoring and execution, as we have shown in this work. The phenotype author only requires knowledge of well documented standards to create a phenotype definition. No knowledge of the data model where the phenotype definition will be executed is required. No access to the actual data by the author is required either, which means that knowledge artifacts developed by third parties can be executed against data while maintaining patient privacy, analogous to the model to data approach used for evaluating machine learning models on healthcare data[147].

Using a formal, unambiguous phenotype definition representation means that existing tools can be reused, and any new conformant tools can be used. As demonstrated here, tools like HAPI FHIR and CQF Ruler can be used out of the box for development and testing. Since many tools exist to generate FHIR-compliant data (e.g., scenario builder*

*http://clinfhir.com/builder.html

and the CQL Testing Framework), they can be leveraged to populate a testing environment that can be used to validate phenotype logic. This testing data can be used with publicly available standards-based testing tools.

The use of a FHIR `Composition` is useful because it allows the individual building blocks of a phenotype (i.e., `Library` and `ValueSet` resources) to be stored and retrieved individually, but also allows the full phenotype definition to be reassembled using the `$document` extended operation. This operation inspects the `Composition` resource and returns a FHIR `Bundle` containing the `Composition` itself, along with all of the required resources to fully specify the phenotype. This is convenient for retrieving and packaging the phenotype definition as a single file to share with other researchers or clinical teams.

While graphical tools for phenotype authoring do exist, for example the OHDSI Atlas cohort creator, the i2b2 query interface, and others[138], they all have limitations the PhEMA Workbench attempts to address. First, the phenotype definitions produced by these tools do not conform to any healthcare standard. This means there is no formal process for making schema changes, which could result in unintentional breaking changes to existing phenotype definitions. Also, while these definitions can be shared between implementations of the same system, they cannot be shared between systems (i.e., they are not cross-platform). Furthermore, the expressivity of the phenotype definitions supported by these tools is limited. For example, these tools may not support the full set of Allen's interval operators, or the wide range of collection and aggregation operations supported by CQL. Even directly using SQL to generate cohorts may not be as convenient as using CQL, since SQL lacks clinical operators such as those used to determine patient age (e.g., current age or age at date of clinical observation), as well as terminology operators to check whether coded values are

part of specific code systems or value sets. Additionally, SQL is very tightly coupled to the data model against which it is executed, which means code is not reusable across databases with different schemas, and the phenotype author must know the target database schema in advance.

Local customization is an important part of EHR-driven phenotyping, as variations in local guidelines and clinical practice can easily result in differences in downstream observational datasets[28]. The PhEMA Workbench supports local customization by allowing users to directly view and edit logic before execution, as well as swap out value sets for ones that are more appropriate for the local context, all while remaining fully standards compliant. Various methods exist to author CQL source code, but only one integrates with a testing environment (the Atom CQL plugin*) and to our knowledge, no other tool integrates with a phenotype repository or has the ability to assemble value sets from various sources.

4.5.1 Limitations

We note the following limitations to this work. First, we only tested a single phenotype definition, and there may be phenotypes for which our approach would not work. For example, if a phenotype exclusively used NLP or machine learning. Additionally, while we demonstrated cross-platform authoring and execution, we only tested a single execution target (OMOP). We previously demonstrated that cross-platform execution is possible[37], but did not evaluate that scenario in this work.

While we did use FHIR as our standard data model, we did not restrict data modeling decisions further by conforming to a FHIR profile. As a result, the FHIR resources and

*https://atom.io/packages/language-cql

fields we chose to use may not exactly match data in other FHIR data repositories, and we cannot communicate our data model computationally to other systems using a FHIR implementation guide.

We make use of the `$cql` extended operation to execute our phenotype definitions during development and testing. This operation could also be used to execute phenotype definitions in production, but the `$cql` operation is not part of the base FHIR specification. The operation is specified in the FHIR Clinical Practice Guidelines implementation guide, which may not be supported by all FHIR servers.

Finally, as we have previously discussed[37], the OMOP execution tool we are using only supports a subset of CQL operators. This tool can be extended to a degree to support additional logical operators, but is fundamentally limited by the expressiveness of the cohort definition representation in OMOP. The CQL language also only supports structured data elements out of the box, but the PhEMA collaborators have developed an NLP integration, CQL4NLP[38], which is yet to be integrated into the Workbench architecture. There is also an upfront cost involved in implementing a CQL engine for a new data platform, which may be prohibitive if engineering resources are not available.

Despite these limitations, we have shown that using a fully FHIR-native phenotype representation is feasible, and enables the decoupling of authoring and execution, leading to several advantages. If such an approach is broadly adopted, it may increase the velocity of biomedical generation by increasing semantically interoperability of phenotype definitions, and facilitating high-throughput automated cohort identification. This will both reduce the time required to collect data for observational studies and the potential for human error.

4.6 Conclusion

We have shown that cross-platform EHR-driven phenotyping, in which a clinician-developed phenotype definition, using a FHIR-native representation, and executed against an OMOP data repository, can achieve results comparable to other methods. Additionally, we have shown that a modular architecture consisting of existing open-source tools, including the newly developed PhEMA Workbench, can provide an effective phenotyping environment that supports users from multiple institutions.

The PhEMA Workbench is a contribution to the phenotyping community that complements existing tools, and requires no changes to existing systems. This gives end users additional options without added restrictions. Furthermore, the formal representation proposed here makes use of the popular, and federally mandated, FHIR standard, which facilitates interoperability, reduces the possibility of ambiguity introduced by human interpretation, and reduces the time required to identify cohorts for biomedical research. All of this should lead to decreased cost and increased velocity of biomedical knowledge generation.

In the future we intend to use the PhEMA Workbench in clinical studies, and are planning to extend the capabilities of the system to include NLP (using CQL4NLP) and more sophisticated machine learning models, which is a central theme of the next phase of the PhEMA grant. We are also planning to continue improving the Workbench, and have a user-centered design study currently underway.

CHAPTER 5

CONCLUSION

In this chapter we will summarize the contributions of this dissertation, present the main conclusions, and discuss the limitations and potential areas of future work.

5.1 Contributions

We began this work in chapter 2 by investigating the nature of phenotype definitions. We extracted a data set of clinically validated phenotype definitions from a mature phenotype repository, and translated them into a formal representation. The correctness of these translations was validated both manually, through a process of code review, and automatically, through the development of a large test suite. We then analyzed the phenotype metadata and definitions both manually and programmatically, and provided the raw data, visualizations, and a synthesized analysis. We characterized terminology usage, logical constructs, and data access patterns. This work produced a data set of computable phenotype definitions that can be used by implementers or by researchers investigating automated execution or formal representations. A manuscript based on chapter 2 is currently at the co-author review stage.

In the second aim (chapter 3) we studied whether a formal representation of phenotype logical criteria based on the nascent CQL standard could facilitate cross-platform phenotyping. While the work done in the first aim demonstrated that CQL is expressive enough to

represent a wide range of phenotype definitions, the work presented in chapter 3 investigated the feasibility of automated execution in multiple environments. We translated a clinically validated phenotype definition into CQL and built a novel tool that enables the execution of CQL-based phenotypes against an OMOP database. We executed the phenotype definition at two large academic medical centers, Northwestern Medicine and Weill Cornell Medicine, and across two data platforms (FHIR and OMOP). Manual reviews were conducted to ensure that logic was correctly applied in all cases. The output of this study was an open-source tool, as well as a publication in the special issue on Human Phenomics of the *Learning Health Systems* journal[37].

In chapter 4 we continued our investigation of standards-based phenotyping by developing and evaluating a tool that supports all phenotyping subtasks. Additionally, we propose a fully FHIR-native phenotype representation and investigate whether this representation enables phenotyping in the existing fragmented phenotyping ecosystem. The developed tool supports authoring, publishing, as well as execution. Features include CQL editing, automated execution in testing and production environments, the assembly of value sets in a variety of formats, interaction with the PheKB API, and packaging phenotype definitions using the proposed FHIR-native representation. We demonstrated how the tool can be used to support the complete phenotyping process by having one author create and publish a FHIR-native phenotype definition at one site, after which two implementers at different sites imported the definition and executed it in their environments using different methods. Validation was done at both sites by manually reviewing a subset of the cohort members to ensure they met the phenotype criteria. The result of this work is a proposed phenotype representation using FHIR and CQL, and an open-source, standards-based phenotyping tool.

The manuscript based on this work is currently being prepared for submission to *JAMIA Open*.

In our original research proposal, we highlighted the challenges faced by researchers using EHR data for secondary purposes. We presented our aims that address these challenges under the framework of the Task-Technology Fit model. Our first aim contributes to the *Task Characteristics* aspect of the model by characterizing the nature of phenotype definitions. Our second and third aims contribute to the *Technology Characteristics* aspect by investigating the feasibility of using CQL and FHIR to formally represent phenotype definitions. The latter two aims also contribute to the *Performance Impacts* aspect of the model by proposing a FHIR-native phenotype representation, developing standards-based tools, and evaluating their performance on the task of EHR-driven phenotyping.

5.2 Conclusions

The phenotype definitions examined in chapter 2 all have the same basic components, namely logical expressions and tabular data. The tabular data consists of lists of codes and constructs to be used for natural language processing (NLP). Phenotype criteria logic can be divided into clinical and operational logic, the latter of which may be impossible to express in a universally correct manner in some cases. This implies that any formal standard used for representing phenotype definitions must have some mechanism to support local customization. Both CQL and FHIR meet this requirement.

We found that the self-reported metadata on PheKB is not very complete, and is in some cases inaccurate. Both of these issues would be addressed by using a formal representation,

from which metadata could be computationally extracted. Additionally, most phenotypes use both structured and unstructured data, so there must either be a preprocessing step in which unstructured data is extracted and converted into a structured format, or phenotype representations must integrate with NLP tools. While almost all phenotypes also use tabular data, this data is usually provided in a non-computable format on PheKB. Furthermore, only a handful of code systems are used extensively and about half of all codes used are ICD-9 codes. Most phenotypes use fewer than 10 value sets to filter data from three or fewer sources, primarily conditions, medications and procedures. Any formal representation must be able to represent both list of codes, and the data from the sources used.

Half of all the analyzed phenotype definitions use fewer than 147 codes and 66 expressions. Only 44 unique types of expressions are used, which is only a small fraction of those supported by CQL (over 200). Total expression depth is less than 20 in most cases, and the depth of data filtering clauses is usually under 12. These measures imply that the phenotype definitions studied are reasonably conceptually simple. There are a few reasons this could be the case. First, even relatively simple phenotype definitions can be difficult to interpret and manually implement, so authors may choose to keep definitions simple to make implementation easier. Another reason could be that EHR-driven phenotyping is a relatively new pursuit in medicine, as clinical trials are still the gold standard. The complexity of phenotype definitions may increase over time as more work is done in this area. Finally, the lack of a formal representation may be a limiting factor, as authors may not have the tools or mechanism to express phenotype criteria in sufficient detail. This may be mitigated as standards like CQL or the next generation of domain-specific knowledge representation standards become more widespread.

Translation of the heart failure (HF) phenotype used in chapter 3 provides further validation that CQL is expressive enough to represent clinically validated phenotype definitions. Although the HF phenotype used numerous expression categories, including aggregate, temporal, and logical, its representation in CQL was straightforward. Additionally, using FHIR as the data model turned out to be a good choice, since a mapping to the OMOP data model was readily available, which simplified implementation of the CQL on OMOP tool. This highlights the fact that using already popular standards results in a network effect that can accelerate knowledge generation.

The conceptual models of the OHDSI Circe (cohort definition) library and CQL differ significantly, and a lossless, bidirectional mapping between the two is not possible to generate programmatically for at least two reasons. First, OHDSI cohort definitions divide criteria into human-defined *inclusion rules* that cannot be automatically inferred. While this limitation does not make translation impossible, it does mean that useful cohort attrition statistics cannot be determined. Second, and more importantly, Circe cohort definitions are simply not as expressive as CQL, so while any criteria specified in Circe can be translated into CQL, the reverse is not true.

The aim 2 work validates our hypothesis that a standards-based representation of phenotype logic can facilitate both automation and portability. Automated execution against a FHIR data store is immediately available when using FHIR and CQL, due to the tools that already exist for these popular standards. Automated execution against the OMOP data model required the development of a novel tool. However, development of this tool was greatly simplified due to the use of existing healthcare standards, which provide the necessary conceptual models, libraries, examples, and expertise.

In distributed study designs, the EHR-driven phenotyping process involves authoring, publishing, and execution. Existing systems only support a subset of these tasks. This fragmentation is further exacerbated by the lack of standards, which means that certain tools cannot be used together without significant effort. In our final study reported in chapter 4, we demonstrate how a fully FHIR-native phenotype definition, along with a suite of standards-compliant tools, can facilitate all the steps involved in the phenotyping process. These tools interoperate with the existing ecosystem of phenotyping tools, achieve excellent results based on manual validation, and can be adopted incrementally without requiring any changes to existing systems.

As we learned in chapter 2, phenotype definitions consist of both tabular data and logic, and for this reason neither CQL nor FHIR alone are enough to fully represent any phenotype definition. We propose a formal representation comprised of a combination of these two standards, which can successfully do so. This representation includes all phenotype criteria logic and value sets, and is packaged using the FHIR-native `Composition` and `Bundle` formats that enable easy assembly and sharing using standard FHIR semantics. The format also enables automated execution and computational analyses like those conducted in chapter 2.

By authoring a thrombotic event phenotype, publishing it to PheKB, and using our tools to execute the phenotype at multiple sites, we validated that our representation does indeed facilitate the full phenotyping process in a heterogenous environment. Implementation time was only limited by database execution time, and the overall time required was orders of magnitude less than the recently reported upper limited of multiple months[45]. Potential for human error was also significantly decreased. This indicates that a standards-based representation with accompanying interoperable tools can increase the velocity of cohort

identification and observational research.

5.3 Limitations

We acknowledge that there are several limitations that may have an impact on our results and conclusions. In our investigation of phenotype variability, we only used phenotype definitions from a single source, and there is no guarantee that the full range of phenotype definition criteria are represented in PheKB. Further, we only analyzed 33 phenotype definitions, which is not a very large sample. There may be logical or metadata constructs not represented in this set. The use of CQL as a logical expression language my not have been the optimal choice. Although we did not encounter any significant challenges using this representation, there may be better alternatives. Similarly, their may be better standards than FHIR for use as a data model and value set representation.

Due to the expressivity of CQL, there are many possible implementations for any given phenotype definition, and while we made every effort to use the simplest implementation we could find, more optimal implementations may exist. Our validation process and automated tests may not cover 100% of edge and corner cases, which means our implementations could contain bugs. Additionally, we did not clinically validate our CQL-based phenotypes, which would further increase confidence in their correctness.

In our aim 2 study we only executed our CQL-based phenotype definition at two sites and against two different data platforms. Execution at additional sites or against additional data platforms may have surfaced errors that our experimental architecture did not.

At least two limitations apply to both our aim 2 and aim 3 studies. First, we only used

a single phenotype definition. While the phenotypes used did include a range of expression categories, they did not include every possible expression. It is possible that different or more complicated phenotype definitions would result in reduced performance. Second, in each study we only manually verified at total of 100 cases and non-cases. Even though we used review selection criteria designed to catch edge cases, we still only manually verified a small percentage of the total number of cohort members identified by the EHR-driven phenotyping process.

Finally, an important limitation that applies to all three research studies presented in this dissertation is our use of structured data and rule-based logic only. Unstructured data is an important source of clinical information, and NLP techniques are widely used in EHR-driven phenotyping. We also did not use any machine learning algorithms, which are an increasingly popular phenotyping method.

5.4 Future Work

There several potential avenues of investigation that could build on the work presented here. In order to gain a deeper understanding of the nature of phenotype definitions, additional phenotypes from PheKB could be analyzed. Phenotype definitions from additional sources could also be analyzed, such as those in other phenotype repositories, both in the United States and abroad, as well as phenotype definitions described in the literature.

In chapter 2 we highlight the fact that some operational logical criteria cannot be specified using universally applicable logical expressions. We propose the idea of using a "system library" approach in which these difficult to generalize criteria have well defined interfaces

that can be locally implemented by participating sites. Further work can be done in this area by enumerating these specific criteria and clearly defining their interfaces.

While we did use clinically validated phenotypes in our work, we did not use our standards-based phenotype representation or tools in clinical research. While our work provides a proof of concept that shows clinical research is feasible, a clear next step is to use our proposed fully FHIR-native phenotype representation and accompanying tools in such studies, and validate the resulting cohorts using clinical chart review.

The Workbench application presented in chapter 4 serves as a minimal viable product that can be used for FHIR-native phenotyping, but there is significant room for improvement. The functionality and user experience can be improved, and the PhEMA project has an ongoing user-centered research study with this objective. Support for additional data platforms and CQL language constructs could be implemented in the CQL on OMOP translator. Potential targets are the i2b2 and PCORnet data platforms.

While we do propose and describe a formal representation using open standards, we do not formally describe this representation in a computable manner using the mechanisms provided by the FHIR standard. A FHIR profile (the means for further constraining the FHIR standard) using `StructureDefinition` and `GraphDefinition` resources could be created. These computable artifacts could then be used to programmatically validate phenotype definitions and test whether software systems are compliant.

5.4.1 Unstructured Data

Finally, it is important that future work addresses the lack of support for unstructured data and machine learning algorithms. Initial work has been done by the PhEMA project team to

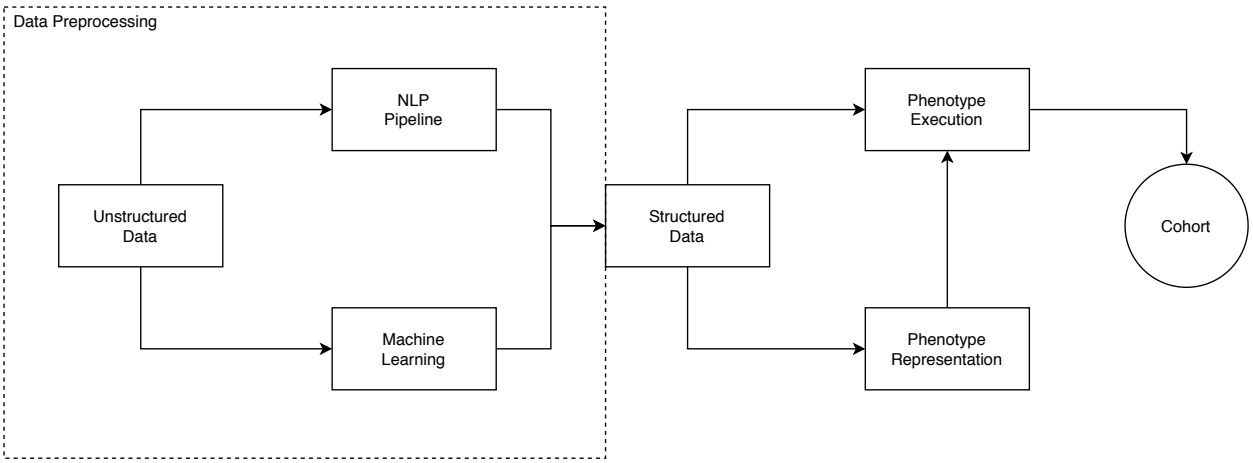


Figure 5.1: Incorporating unstructured data using a preprocessing step.

integrate NLP tools with both FHIR[27] and CQL[38], but these methods are not yet supported by the Workbench application or incorporated into our FHIR-native representation. We are not aware of any work that proposes a formal phenotype representation that includes integration with machine learning.

Figure 5.1 illustrates how NLP and machine learning algorithms can be incorporated into the methods described in this work by using a data preprocessing step. Unstructured data such as clinical notes can be used as input for NLP pipelines, which ultimately produce structured output that can be used as input by our tools and methods. Similarly, other unstructured data such as image or sensor data could be processed by machine learning algorithms to produce structured output.

An important consideration if this technique is to be successful is the output data format produced by the preprocessing step. Either this format must exactly match that of the existing structured data, or the existing format must be extended to support the newly produced data. The latter approach was used in the initial work by the PhEMA team cited above, and was implemented using the standardized FHIR extension mechanism.

5.5 Final Remarks

The overarching goal of the work presented in this dissertation was to advance the methods used for EHR-driven phenotyping, with the hope that this would accelerate the velocity at which EHR data can be used to generate biomedical knowledge. We have made contributions towards this goal in each of the three research studies conducted. First, we provided insight into the nature of a clinically validated set of phenotype definitions. We then demonstrated how the CQL standard can be used to facilitate cross-platform EHR-driven phenotyping by developing a novel execution engine. We extended this work in the third study by proposing a fully standards-based phenotype representation using FHIR and CQL, and evaluating an interoperable, incrementally adoptable phenotyping tool that uses this representation. These three studies make incremental but significant contributions towards methods that may evolve to ultimately support high-throughput biomedical knowledge generation.

APPENDIX A

CQL ON OMOP DESIGN CONSIDERATIONS

The CQL on OMOP translator takes CQL or ELM (parsed CQL) as input, and generates an OHDSI cohort definition as output. The functionality also exists to submit the cohort definition to the OHDSI Web API to create the cohort definition, initiate the cohort generation job, and poll for the cohort results.

As this is a PhEMA project, the ultimate goal of this work is to evaluate whether the CQL language can be used for cross-platform EHR-driven phenotyping.

A.1 Circe Overview

The library used internally by the OHDSI Web API to represent and execute cohort definitions is called *Circe*. The normal way that users create cohort definitions is by using the OHDSI web interface, called *Atlas*. The user creates a list of inclusion rules by clicking buttons and selecting from dropdown menus. To each inclusion rule they add criteria groups, and to each group they add specific domain criteria by using standard user interface controls.

Once saved, the Atlas application generates a JSON representation of the cohort definition, which is then submitted to the Web API, which saves it to the OHDSI database. The user must then take a separate action to initiate the cohort generation job. As part of this asynchronous job, the Circe library deserializes the JSON version of the cohort definition into Java objects that it uses internally. Circe then uses a set of builder classes, along with

SQL templates to construct the database queries needed to generate the cohort from the cohort definition.

There are several benefits to this approach. First, the graphical user interface allows users without any knowledge of SQL, and only minimal knowledge of the OMOP Common Data Model (CDM), to construct cohorts of clinical or research interest. Separation of logic into distinct inclusion rules allows for more efficient database queries, since successive inclusion rules are only applied to the results of the previous rule. Separate inclusion rules further allow for the generation of attrition statistics and visualizations. Finally, using SQL templates enables the creation of manually optimized queries that may be better than those generated by an object-relational mapper.

The Circe UML diagram (fig A.1) shows how cohort expressions are assembled, as well as what type of criteria exist.

The Clinical Quality Language (CQL) is a domain-specific language focused on the clinical quality and decision support domains. It is parsed into a canonical abstract syntax tree they call the Expression Logical Model (ELM). Libraries exist to parse CQL into the equivalent ELM representation, so internally all our computation is done on ELM.

CQL is a highly expressive language approaching the complexity of a general-purpose programming language. As such, it is able to represent significantly more constructs than Circe, as can be seen in the ELM UML diagram (figure A.2). One simple example is that CQL is able to evaluate the expression 1 + 1 and return the result of 2. Furthermore, there are usually many different ways that the same logic can be represented in CQL. CQL is also data model independent, which means that libraries must specify the data for which they are written, and the evaluation engine must know about this data model in order to evaluate

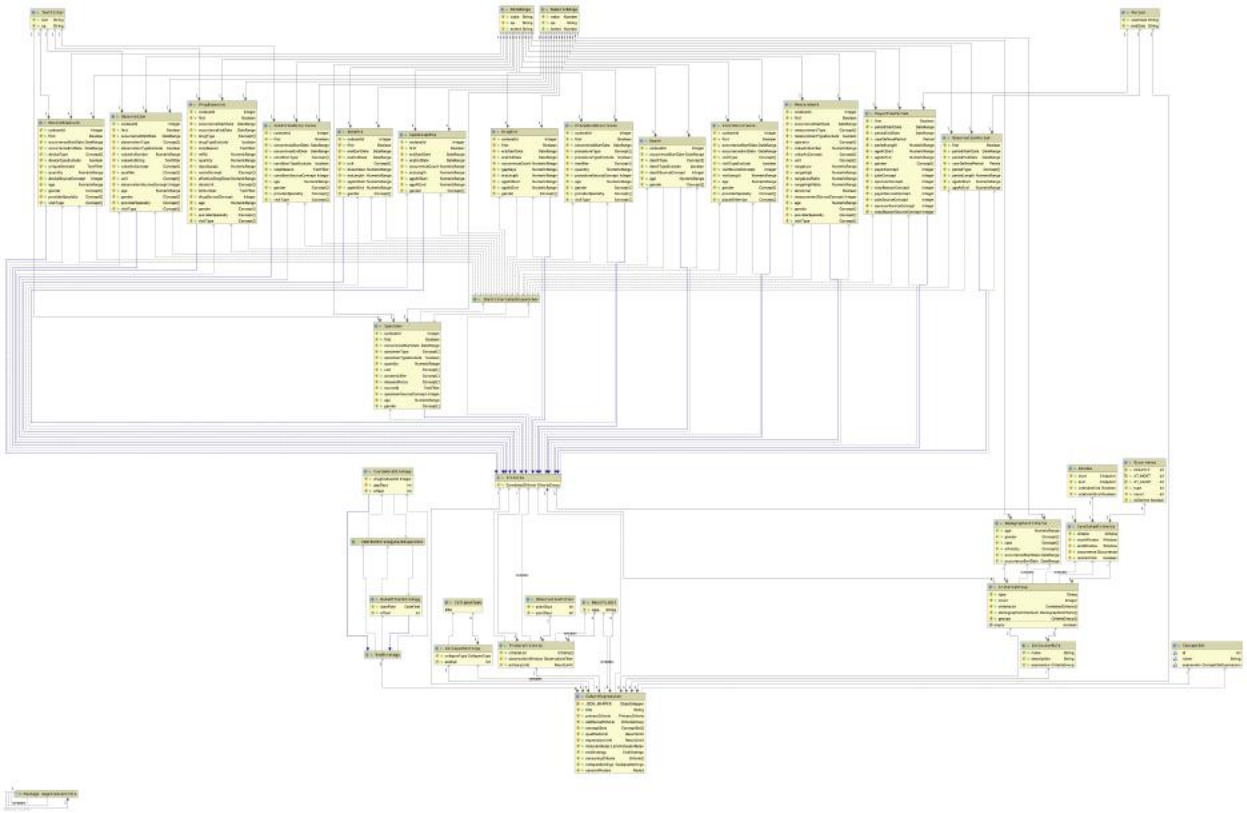


Figure A.1: Circe UML diagram. View online at https://github.com/PheMA/cql-on-omop/blob/master/docs/img/circe-uml.png

the library.

A CQL library can contain many statements, each of which is evaluated separately (although statements can reference each other). This means that evaluating a CQL library returns multiple results - one for each statement in the library. Further, each result can be one of many different data types.

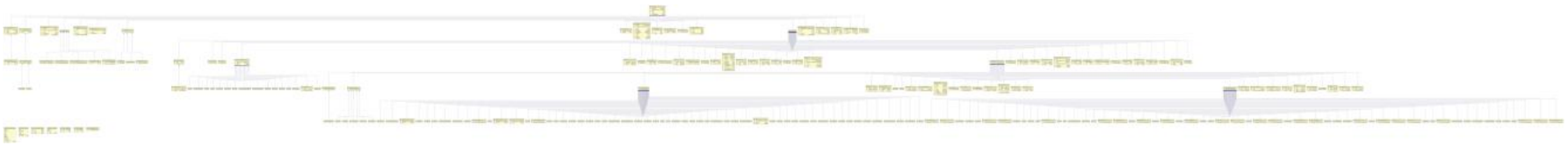


Figure A.2: ELM UML diagram. View online at https://github.com/PheMA/cql-on-omop/blob/master/docs/img/elm-uml.png.

A.2 Implementation Considerations

A.2.1 Language Support

For the above reasons, we can only support translating a limited subset of the CQL language. We therefore define a set of supported language constructs, along with some conventions that must be followed so that we can successfully translate the CQL library to a Circe cohort definition. We aim to support the following language constructs:

- The `CalculateAge()` function, used to determine the age of the patient [docs]
- Simple `retrieve` operation with terminology filtering, to access the underlying data [docs]
- The following numeric comparisons: $=$, $<$, $\leq$, $>$, $\geq$ [docs]
- Some `query` operations with `where` clause filtering [docs]†
- Some correlated `query` operations with a single relationship [docs]†
- Timing relationships in `query` constructs (including correlated queries) [docs]†
- The `and` and `or` logical operators [docs]

† The `query` operations that we are able to support is limited by the criteria and criteria attributes that Circe is able to represent.

More operations will be added to the above list over time, for example, additional demographic characteristics.

A.2.2 Data Model

A simple approach would be to use the OMOP CDM as the data model in our CQL libraries, but this would limit the environments in which the library can be executed. We have therefore taken the decision to use the QUICK data model*, which is a set of FHIR profiles and data type mappings that are focused on quality measurement and decision support use cases. It is likely that many CQL libraries will be written using the QUICK data model, and supporting this data model means that we are able to re-use logic written for many clinical quality measurement and decision support use cases.

In order to map the QUICK model references to the OMOP CDM, we use the mappings published by the Common Data Model Harmonization project[46].

A.2.3 Conventions

A.2.3.1 `Patient` *Context Only*

CQL libraries may contain zero or more `context` statements. This statement tells the interpreter to potentially apply some data filtering. For example, if the `Patient` context is specified, then only data for a specific patient is included in the evaluation. If the `Unfiltered` context is used, then data for all patients is considered. Data models may optionally specify additional evaluation contexts [docs].

We support only the `Patient` context, which means that each statement should be written with knowledge that it will be applied to a single patient only. This also means that we cannot support the population-based aggregate operators [docs].

*We were thinking of using QUICK when this was written, but subsequently switched to base FHIR.

A.2.3.2 Phenotype Entry Point Statement

Since a CQL library may contain multiple statements, but we only create one Circe cohort definition, we must somehow determine which statement represents the phenotype definition. Currently, the translator is written in such a way that it takes the name of the phenotype definition statement as a parameter. An alternative approach could be to use a statement naming convention, or some other way to annotate the correct statement definition.

A.2.3.3 Boolean Return Types

The ultimate decision that must be made for each patient is whether or not they should be included in the cohort specified by the Circe cohort definition, or equivalent CQL library. In Circe, this decision is made by taking the logical conjunction of each inclusion rule applied to each patient.

The approach we have taken is to require that all CQL statements return a Boolean value. At first this may seem limiting, but it actually exactly matches how Circe represents cohort definitions. Each Circe criteria determines exactly the Boolean result corresponding to whether or not a given patient meets the criteria.

A.3 Implementation Details

Some technical implementation details are described below.

A.3.1 Inclusion Rules

When users create cohort definitions using Atlas, it is convenient to group conceptually similar criteria together in a single inclusion rule. One example is that two demographic criteria,

such as one for age and for gender may both be added to one inclusion rule. Without introducing additional conventions, it is unfortunately not possible to detect these conceptually similar criteria in the translator code. As a result, all criteria are added to a single inclusion rule.

There are two unfortunately consequences. First, it not possible to determine the attrition contribution from groups of criteria. Second, this limitation may result in poorer performing database queries, since criteria are not applied only to the results of preceding inclusion rules.

It may be worth introducing additional conventions to overcome these limitations.

A.3.2 Criteria Groups, Correlated Criteria, and Criteria

In Circe, criteria groups are used to group collections of criteria together. Criteria groups must also specify how the contained criteria should apply. For example, the user can specify whether all criteria must apply, whether any one can apply, or a whether minimum or maximum number of criteria must apply.

Criteria themselves can be correlated or uncorrelated. For example, looking for a condition that matches a specific value set is an uncorrelated criteria. Looking for a measurement with a specific value that occurs within some time frame of a specific procedure is an example of a correlated criteria.

Unfortunately, in version `1.7.0` of the Circe library, criteria groups can only contain instances of the `CorelatedCriteria` class (or `DemographicCriteria` or other `CriteriaGroupss`). This means that specific domain criteria (e.g. `ProcedureOccurrence`) must always be wrapped in a `CorelatedCriteria`, even when uncorrelated. Further, the `Criteria` par-

ent class of all domain criteria has a field called `CorelatedCriteria`, which is actually of type `CriteriaGroups`, which can be very confusing. However, this field does determine which criteria groups are correlated to the specific domain criteria instance.

Consider the very simple case of a cohort definition where we are only looking for patients that have had a procedure matching a specific value set. To accomplish this, we would create a `CohortDefinition` instance, to which we would add a `CohortExpression` containing a single `InclusionRule`. The `InclusionRule` class contains a single expression member that is of type `CriteriaGroups`.

To construct the logic, we begin by creating an instance of the `ProcedureOccurrence` criteria referencing the appropriate value set (more on value sets below). We leave the `CorelatedCriteria` field (note: this is the name of the field, not its type, which is actually `CriteriaGroups`) `null`, since this is an uncorrelated criteria.

We must then create a `CorelatedCriteria`, and set the criteria field to the `ProcedureOccurrence` instance just created. Finally, we can add this CorelatedCriteria to a `CriteriaGroups`, which we can then add to the `InclusionRule`. We end up with something that looks like the following:

```
• CohortDefinition
    • CohortExpression
       • List<InclusionRule>
          - 0: CriteriaGroup
                • List<CorelatedCriteria>
                   - 0: CorelatedCriteria
                         • ProcedureOccurrence
                            • CriteriaGroup = null (name: CorelatedCriteria, type: CriteriaGroup)
```

One reason why all `Criteria` must be wrapped in a `CorelatedCriteria` is because cohorts in OHDSI are modeled using the idea of cohort entry event, and all criteria are

Class	Description
`CohortDefinition`	This is actually a class in the Web API, not Circe, but it is the outer most wrapper of the payload that is sent to the Web API.
`CohortExpression`	This class is the container for the expression logic, including the `InclusionRule` instances, and the cohort entry event (an instance of `PrimaryCriteria`).
`InclusionRule`	An inclusion rule just wraps a single `CriteriaGroup`, giving it a name.
`CriteriaGroup`	A criteria group contains any number of `CorelatedCriteria`, `DemographicCriteria` and/or other `CriteriaGroup` instances. It also specifies how these criteria are applied (e.g., `ALL`, `ANY`, `AT_LEAST 3`, etc).
`CorelatedCriteria`	This class wraps all domain criteria, and associates start and end windows with them (relative to the parent criteria or entry event). This class also has an `Occurence` field specifying how many, say, procedures should be found.
`Criteria`	This abstract class is the parent of all the domain criteria (e.g., `ConditionOccurrence`, `Observation`, etc). Criteria also contains a field (unfortunately) called `CorelatedCriteria` of type `CriteriaGroup` which facilitates temporal correlation between criteria.
`DemographicCriteria`	This is a special type of criteria allowing the user to filter cohort members based on demographic characteristics.

Table A.1: Short summary of Circe classes.

actually correlated in some way to this entry event, either directly or indirectly. The `CorelatedCriteria` class therefore has `startWindow` and `endWindow` fields, which are relative (directly or indirectly) to the entry event.

Finally, `CorelatedCriteria` also has an `occurence` field of type `Occurence`, which is used to describe how domain criteria should apply. Continuing the above example, if the procedure should occur at least three times, then we specify this using an `Occurence` instance.

Note that in all cases except for Boolean logic (see below), we translate CQL constructs to their equivalent `CorelatedCriteria` representations.

Table A.1 provides a short summary of some of the important Circe classes.

A.3.3 Nested Boolean Logic

Boolean logic can only be represented in Circe using `CriteriaGroup` instances of type `ALL` and `ANY`, representing Boolean `and` and `or` statements respectively. We support arbitrarily nested Boolean `and` and `or` statements, and implement such nesting using nested `CriteriaGroup` instances.

A.4 Value Sets

Circe uses the `ConceptSet` class to represent value sets. All referenced value sets are included inline in the `CohortExpression` instance. I believe this is so that descendents, mapped, and excluded concepts specified in existing OHDSI concept sets can all be resolved ahead of evaluating the cohort expression.

That said, our implementation does not make use of existing concept sets. Instead, we define a service interface used to retrieve the relevant concepts sets. We have service implementations that read PhEMA value sets from CSV files, and resolve concepts using the OHDSI Web API. This supports using publicly accessible value sets based on standard terminologies, and decouples the implementation from the OHDSI platform.

We also have a service implementation that reads pre-resolved concept sets from file in JSON format. This is more efficient, especially for large concept sets, since concepts must otherwise be resolved one at a time by performing a search using the Web API, which is an expensive operation.

A.5 Alternative Approaches

The approach described above is not the only possibility. Another approach would be to extend the reference implementation of the CQL engine (or create a new implementation) to directly support the OMOP CDM as a data model. The advantage of this approach is that the CQL library author would have full access to the expressiveness of the CQL language, and could write queries of any type, for any purpose, not just phenotyping.

The downside of this approach is that CQL libraries written against this data model would then be tied to the OHDSI platform, and would not be cross-platform, as in the current implementation. Further, the full set of OHDSI tools for cohort analysis would no longer be available to the user. Importantly, in the current approach, a user can inspect the generated cohort definition using the existing Atlas interface to manually confirm whether or not the logic is correct, which would not be possible in a pure CQL data provider implementation.

REFERENCES

[1] Archibald Leman Cochrane and Others. *Effectiveness and efficiency: random reflections on health services*, volume 900574178. Nuffield Provincial Hospitals Trust London, 1972.

[2] Stuart L. Silverman. From Randomized Controlled Trials to Observational Studies. *American Journal of Medicine*, 122(2):114–120, 2009. ISSN 00029343. doi: 10.1016/j.amjmed.2008.09.030. URL http://dx.doi.org/10.1016/j.amjmed.2008.09.030.

[3] Institute of Medicine (IOM). *The Learning Healthcare System*. National Academies Press, Washington, D.C., jun 2007. ISBN 978-0-309-10300-8. doi: 10.17226/11903. URL http://www.ncbi.nlm.nih.gov/books/NBK53488/http://www.nap.edu/catalog/11903.

[4] The Office of the National Coordinator for Health Information Technology (ONC). Percent of Hospitals, By Type, that Possess Certified Health IT [Accessed: 2019-05-10]. URL https://dashboard.healthit.gov/quickstats/pages/certified-electronic-health-record-technology-in-hospitals.php.

[5] Ronald Margolis, Leslie Derr, Michelle Dunn, et al. The National Institutes of Health's big data to knowledge (BD2K) initiative: Capitalizing on biomedical big data. *Journal of the American Medical Informatics Association*, 21(6):957–958, 2014. ISSN 1527974X. doi: 10.1136/amiajnl-2014-002974.

[6] NIH. National Institutes of Health (NIH) — All of Us [Accessed 2019-05-14]. URL https://allofus.nih.gov/.

[7] Francis S. Collins and Harold Varmus. A New Initiative on Precision Medicine. *New England Journal of Medicine*, 372(9):793–795, feb 2015. ISSN 0028-4793. doi: 10.1056/NEJMp1500523. URL http://www.nejm.org/doi/10.1056/NEJMp1500523.

[8] Juan M Banda, Martin Seneviratne, Tina Hernandez-Boussard, and Nigam H Shah. Advances in Electronic Phenotyping: From Rule-Based Definitions to Machine Learning Models. *Annual Review of Biomedical Data Science*, 2018. doi: 10.1146/annurev-biodatasci. URL https://doi.org/10.1146/annurev-biodatasci-080917-013315www.annualreviews.org.

[9] Zoë Slote Morris, Steven Wooding, and Jonathan Grant. The answer is 17 years, what is the question: understanding time lags in translational research. *Journal of the Royal Society of Medicine*, 104(12):510–520, 2011. ISSN 1758-1095. doi: 10.1258/jrsm.2011.110180. URL http://jrs.sagepub.com/content/104/12/510.full.

[10] Mary Regina Boland, George Hripcsak, Yufeng Shen, et al. Defining a comprehensive verotype using electronic health records for personalized medicine. *Journal of the American Medical Informatics Association*, 20(e2):e232–e238, dec 2013. ISSN 1067-5027. doi: 10.1136/amiajnl-2013-001932. URL https://academic.oup.com/jamia/article-lookup/doi/10.1136/amiajnl-2013-001932.

[11] Adam B. Wilcox. Leveraging Electronic Health Records for Phenotyping. In Philip R.O. Payne and Peter J. Embi, editors, *Translational Informatics: Realizing the Promise of Knowledge-Driven Healthcare*, Health Informatics, pages 61–74. Springer London, London, 2015. ISBN 978-1-4471-4645-2. doi: 10.1007/978-1-4471-4646-9_4. URL http://link.springer.com/10.1007/978-1-4471-4646-9{_}4http://link.springer.com/10.1007/978-1-4471-4646-9.

[12] Catherine A. McCarty, Rex L. Chisholm, Christopher G. Chute, et al. The eMERGE Network: A consortium of biorepositories linked to electronic medical records data for conducting genomic studies. *BMC Medical Genomics*, 4(1):13, 2011. ISSN 17558794. doi: 10.1186/1755-8794-4-13. URL http://www.biomedcentral.com/1755-8794/4/13.

[13] Omri Gottesman, Helena Kuivaniemi, Gerard Tromp, et al. The Electronic Medical Records and Genomics (eMERGE) Network: Past, present, and future. *Genetics in Medicine*, 15(10):761–771, 2013. ISSN 10983600. doi: 10.1038/gim.2013.72.

[14] Hana Zouk, Eric Venner, Niall J. Lennon, et al. Harmonizing Clinical Sequencing and Interpretation for the eMERGE III Network. *American Journal of Human Genetics*, 105(3):588–605, 2019. ISSN 15376605. doi: 10.1016/j.ajhg.2019.07.018.

[15] Robert M Califf. The Patient-Centered Outcomes Research Network: a national infrastructure for comparative effectiveness research. *North Carolina medical journal*, 75(3):204–10, 2004. ISSN 0029-2559. URL http://www.ncbi.nlm.nih.gov/pubmed/24830497.

[16] Huan Mo, William K. Thompson, Luke V. Rasmussen, et al. Desiderata for computable representations of electronic health records-driven phenotype algorithms. *Journal of the American Medical Informatics Association*, 22(6):1220–1230, 2015. ISSN 1527974X. doi: 10.1093/jamia/ocv112.

[17] ACT Network Homepage. URL http://www.actnetwork.us/national.

[18] George Hripcsak, Jon D. Duke, Nigam H. Shah, et al. Observational Health Data Sciences and Informatics (OHDSI): Opportunities for Observational Researchers. *Studies in Health Technology and Informatics*, 216:574–578, 2015. ISSN 18798365. doi: 10.3233/978-1-61499-564-7-574.

[19] Jennifer A Pacheco, Luke V Rasmussen, Richard C Kiefer, et al. A case study evaluating the portability of an executable computable phenotype algorithm across multiple institutions and electronic health record environments. *Journal of the*

American Medical Informatics Association : JAMIA, 0(September):1–7, 2018. ISSN 1067-5027. doi: 10.1093/jamia/ocy101. URL http://doi.org/10.1093/jamia/ocy101.

[20] Phenotype Modeling and Execution Architecture. URL http://informatics.mayo.edu/phema/index.php/Main{_}Page.

[21] Mike Conway, Richard L Berg, David Carrell, et al. Analyzing the heterogeneity and complexity of Electronic Health Record oriented phenotyping algorithms. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2011:274–83, 2011. ISSN 1942-597X. doi: PMC3243189. URL http://www.ncbi.nlm.nih.gov/pubmed/22195079.

[22] Kevin J. Peterson and Jyotishman Pathak. Scalable and High-Throughput Execution of Clinical Quality Measures from Electronic Health Records using MapReduce and the JBoss® Drools Engine. *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, 2014:1864–1873, 2014. ISSN 1942597X.

[23] Jyotishman Pathak, Kent R. Bailey, Calvin E. Beebe, et al. Normalization and standardization of electronic health records for high-throughput phenotyping: The sharpn consortium. *Journal of the American Medical Informatics Association*, 20 (E2):341–348, 2013. ISSN 10675027. doi: 10.1136/amiajnl-2013-001939.

[24] Huan Mo, Jennifer A Pacheco, Luke V Rasmussen, et al. A Prototype for Executable and Portable Electronic Clinical Quality Measures Using the KNIME Analytics Platform. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2015(Icd):127–31, 2015. ISSN 2153-4063. URL http://www.ncbi.nlm.nih.gov/pubmed/26306254.

[25] Huan Mo, Guoqian Jiang, Jennifer A Pacheco, et al. A Decompositional Approach to Executing Quality Data Model Algorithms on the i2b2 Platform. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2016:167–75, 2016. ISSN 2153-4063. URL http://www.ncbi.nlm.nih.gov/pubmed/27570665.

[26] Martin Chapman, Luke V. Rasmussen, Jennifer A. Pacheco, and Vasa Curcin. Phenoflow: A Microservice Architecture for Portable Workflow-based Phenotype Definitions. *medRxiv*, 2020. doi: 10.1101/2020.07.01.20144196.

[27] Na Hong, Andrew Wen, Daniel J. Stone, et al. Developing a FHIR-based EHR phenotyping framework: A case study for identification of patients with obesity and multiple comorbidities from discharge summaries. *Journal of Biomedical Informatics*, 99(April):103310, 2019. ISSN 15320464. doi: 10.1016/j.jbi.2019.103310. URL https://doi.org/10.1016/j.jbi.2019.103310.

[28] Luke V Rasmussen, Pascal S Brandt, Guoqian Jiang, et al. Considerations for Improving the Portability of Electronic Health Record-Based Phenotype Algorithms. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2019:755–764, 2019.

ISSN 1942-597X. URL http://www.ncbi.nlm.nih.gov/pubmed/32308871http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC7153055.

[29] Prakash Adekkanattu, Guoqian Jiang, Yuan Luo, et al. Evaluating the Portability of an NLP System for Processing Echocardiograms: A Retrospective, Multi-site Observational Study. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2019:190–199, 2019. ISSN 1942-597X. URL /pmc/articles/PMC7153064//pmc/articles/PMC7153064/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC7153064/http://www.ncbi.nlm.nih.gov/pubmed/32308812http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC7153064.

[30] Dale L. Goodhue and Ronald L. Thompson. Task-Technology Fit and Individual Performance. *MIS Quarterly*, 19(2):213, jun 1995. ISSN 02767783. doi: 10.1007/978-1-4419-6108-2_5. URL https://www.jstor.org/stable/249689?origin=crossref.

[31] Rachael L. Fleurence, Lesley H. Curtis, Robert M. Califf, et al. Launching PCORnet, a national patient-centered clinical research network. *Journal of the American Medical Informatics Association*, 21(4):578–582, 2014. ISSN 1527974X. doi: 10.1136/amiajnl-2014-002747.

[32] Jacqueline C. Kirby, Peter Speltz, Luke V. Rasmussen, et al. PheKB: A catalog and workflow for creating electronic phenotype algorithms for transportability. *Journal of the American Medical Informatics Association*, 23(6):1046–1052, 2016. ISSN 1527974X. doi: 10.1093/jamia/ocv202.

[33] Jyotishman Pathak, Abel N Kho, and Joshua C Denny. Electronic health records-driven phenotyping: challenges, recent advances, and perspectives. *Journal of the American Medical Informatics Association*, 20(e2):e206–e211, dec 2013. ISSN 1067-5027. doi: 10.1136/amiajnl-2013-002428. URL https://academic.oup.com/jamia/article-lookup/doi/10.1136/amiajnl-2013-002428.

[34] Katherine M. Newton, Peggy L. Peissig, Abel Ngo Kho, et al. Validation of electronic medical record-based phenotyping algorithms: Results and lessons learned from the eMERGE network. *Journal of the American Medical Informatics Association*, 20 (E1), 2013. ISSN 10675027. doi: 10.1136/amiajnl-2012-000896. URL https://pubmed.ncbi.nlm.nih.gov/23531748/.

[35] Chaitanya Shivade, Preethi Raghavan, Eric Fosler-Lussier, et al. A review of approaches to identifying patient phenotype cohorts using electronic health records. *Journal of the American Medical Informatics Association : JAMIA*, 21(2):221–30, 2014. ISSN 1527-974X. doi: 10.1136/amiajnl-2013-001935. URL /pmc/articles/PMC3932460//pmc/articles/PMC3932460/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC3932460/http://www.ncbi.nlm.nih.gov/pubmed/24201027http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC3932460.

[36] Guoqian Jiang, Eric Prud'Hommeaux, Guohui Xiao, and Harold R. Solbrig. Developing A Semantic Web-based Framework for Executing the Clinical Quality Language Using FHIR. *CEUR Workshop Proceedings*, 2042:1–5, 2017. ISSN 16130073.

[37] Pascal S. Brandt, Richard C. Kiefer, Jennifer A. Pacheco, et al. Toward cross-platform electronic health record-driven phenotyping using Clinical Quality Language. *Learning Health Systems*, 4(4):1–9, 2020. ISSN 23796146. doi: 10.1002/lrh2.10233.

[38] Andrew Wen, Luke V Rasmussen, Daniel Stone, et al. CQL4NLP : Development and Integration of FHIR NLP Extensions in Clinical Quality Language for EHR-driven Phenotyping. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2021:(In Press).

[39] Harriette G. C. Van Spall, Andrew Toren, Alex Kiss, and Robert A. Fowler. Eligibility Criteria of Randomized Controlled Trials Published in High-Impact General Medical Journals. *JAMA*, 297(11):1233, mar 2007. ISSN 0098-7484. doi: 10.1001/jama.297.11.1233. URL https://jamanetwork.com/journals/jama/article-abstract/206151.

[40] Jessica Ross, Samson Tu, Simona Carini, and Ida Sim. Analysis of eligibility criteria complexity in clinical trials. *Summit on translational bioinformatics*, 2010:46–50, mar 2010. ISSN 2153-6430. URL http://www.ncbi.nlm.nih.gov/pubmed/21347148.

[41] Chunhua Weng, Samson W. Tu, Ida Sim, and Rachel Richesson. Formal representation of eligibility criteria: A literature review. *Journal of Biomedical Informatics*, 43(3):451–467, 2010. ISSN 15320464. doi: 10.1016/j.jbi.2009.12.004.

[42] Rachel L. Richesson, W. Ed Hammond, Meredith Nahm, et al. Electronic health records based phenotyping in next-generation clinical trials: A perspective from the NIH health care systems collaboratory. *Journal of the American Medical Informatics Association*, 20(E2), 2013. ISSN 10675027. doi: 10.1136/amiajnl-2013-001926.

[43] David A Dorr, Aaron M Cohen, Marsha Pierre-Jacques Williams, et al. From simply inaccurate to complex and inaccurate: complexity in standards-based quality measures. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2011:331–8, 2011. ISSN 1942-597X. URL http://www.ncbi.nlm.nih.gov/pubmed/22195085.

[44] William K Thompson, Luke V Rasmussen, Jennifer A Pacheco, et al. An evaluation of the NQF Quality Data Model for representing Electronic Health Record driven phenotyping algorithms. *AMIA Annual Symposium proceedings*, 2012:911–20, 2012. ISSN 1942-597X. URL http://www.ncbi.nlm.nih.gov/pubmed/23304366.

[45] Ning Shang, Cong Liu, Luke V. Rasmussen, et al. Making work visible for electronic phenotype implementation: Lessons learned from the eMERGE network. *Journal of Biomedical Informatics*, 99(August):103293, 2019. ISSN 15320464. doi: 10.1016/j.jbi.2019.103293. URL https://doi.org/10.1016/j.jbi.2019.103293.

[46] HL7 International-Biomedical Research, and Regulation Work Group. Common Data Models Harmonization. URL https://build.fhir.org/ig/HL7/cdmh/cdmh-overview.html.

[47] Olivier Bodenreider, Duc Nguyen, Pishing Chiang, et al. The NLM value set authority center. *Studies in health technology and informatics*, 192:1224, 2013. ISSN 1879-8365. URL http://www.ncbi.nlm.nih.gov/pubmed/23920998.

[48] Christian Gulden, Sebastian Mate, Hans-Ulrich Prokosch, and Stefan Kraus. Investigating the Capabilities of FHIR Search for Clinical Trial Phenotyping. *Studies in health technology and informatics*, 253:3–7, 2018. ISSN 1879-8365. doi: 10.3233/978-1-61499-896-9-3. URL http://www.ncbi.nlm.nih.gov/pubmed/30147028.

[49] Md Rezaul Karim, Binh-Phi Nguyen, Lukas Zimmermann, et al. A Distributed Analytics Platform to Execute FHIR-based Phenotyping Algorithms. Technical report, dec 2018. URL http://www.hl7.org/implement/standards/.

[50] Frank A. Meineke, Sebastian Stäubert, Matthias Löbe, et al. Design and Concept of the SMITH Phenotyping Pipeline. *Studies in health technology and informatics*, 267: 164–172, sep 2019. ISSN 1879-8365. doi: 10.3233/SHTI190821. URL http://www.ncbi.nlm.nih.gov/pubmed/31483269.

[51] Robert C. McClure, Caroline L. Macumber, Julia L. Skapik, and Anne Marie Smith. Igniting Harmonized Digital Clinical Quality Measurement through Terminology, CQL, and FHIR. *Applied Clinical Informatics*, 11(1):23–33, 2020. ISSN 18690327. doi: 10.1055/s-0039-3402755. URL /pmc/articles/PMC6949169//pmc/articles/PMC6949169/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC6949169/.

[52] Terence Parr. *Language implementation patterns: create your own domain-specific and general programming languages*. Pragmatic Bookshelf, 2009.

[53] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Longman Publishing Co., Inc., USA, 1995. ISBN 0201633612. doi: 10.5555/186897.

[54] Amber Dahlin, Joshua Denny, Dan M Roden, et al. CMTR1 is associated with increased asthma exacerbations in patients taking inhaled corticosteroids. *Immunity, inflammation and disease*, 3(4):350–359, jul 2015. ISSN 2050-4527. doi: 10.1002/iid3.73. URL https://pubmed.ncbi.nlm.nih.gov/26734457https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4693729/.

[55] Marylyn D. Ritchie, Joshua C. Denny, Dana C. Crawford, et al. Robust Replication of Genotype-Phenotype Associations across Multiple Diseases in an Electronic Medical Record. *American Journal of Human Genetics*, 86(4):560–572, 2010. ISSN 00029297. doi: 10.1016/j.ajhg.2010.03.003. URL http://dx.doi.org/10.1016/j.ajhg.2010.03.003https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2850440/pdf/main.pdf.

[56] Todd Lingren, Pei Chen, Joseph Bochenek, et al. Electronic Health Record Based Algorithm to Identify Patients with Autism Spectrum Disorder. *PloS one*, 11(7): e0159621–e0159621, jul 2016. ISSN 1932-6203. doi: 10.1371/journal.pone.0159621. URL https://pubmed.ncbi.nlm.nih.gov/27472449https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4966969/.

[57] Jean Coquet, Selen Bozkurt, Kathleen M Kan, et al. Comparison of orthogonal NLP methods for clinical phenotyping and assessment of bone scan utilization among prostate cancer patients. *Journal of biomedical informatics*, 94:103184, jun 2019. ISSN 1532-0480. doi: 10.1016/j.jbi.2019.103184. URL https://pubmed.ncbi.nlm.nih.gov/31014980https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6584041/.

[58] Andrea H Ramirez, Jonathan S Schildcrout, Dana L Blakemore, et al. Modulators of normal electrocardiographic intervals identified in a large electronic medical record. *Heart rhythm*, 8(2):271–277, feb 2011. ISSN 1556-3871. doi: 10.1016/j.hrthm.2010.10.034. URL https://pubmed.ncbi.nlm.nih.gov/21044898https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3057498/.

[59] Joshua C Denny, Marylyn D Ritchie, Dana C Crawford, et al. Identification of genomic predictors of atrioventricular conduction: using electronic medical records as a tool for genome science. *Circulation*, 122(20):2016–2021, nov 2010. ISSN 1524-4539. doi: 10.1161/CIRCULATIONAHA.110.948828. URL https://pubmed.ncbi.nlm.nih.gov/21041692https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2991609/.

[60] Marylyn D Ritchie, Joshua C Denny, Rebecca L Zuvich, et al. Genome- and phenome-wide analyses of cardiac conduction identifies markers of arrhythmia risk. *Circulation*, 127(13):1377–1385, apr 2013. ISSN 1524-4539. doi: 10.1161/CIRCULATIONAHA.112.000604. URL https://pubmed.ncbi.nlm.nih.gov/23463857https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3713791/.

[61] Janina M Jeff, Marylyn D Ritchie, Joshua C Denny, et al. Generalization of variants identified by genome-wide association studies for electrocardiographic traits in African Americans. *Annals of human genetics*, 77(4):321–332, jul 2013. ISSN 1469-1809. doi: 10.1111/ahg.12023. URL https://pubmed.ncbi.nlm.nih.gov/23534349https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3743946/.

[62] Peggy L Peissig, Luke V Rasmussen, Richard L Berg, et al. Importance of multi-modal approaches to effectively identify cataract cases from electronic health records. *Journal of the American Medical Informatics Association : JAMIA*, 19(2): 225–234, 2012. ISSN 1527-974X. doi: 10.1136/amiajnl-2011-000456. URL https://pubmed.ncbi.nlm.nih.gov/22319176https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3277618/.

[63] Carol J Waudby, Richard L Berg, James G Linneman, et al. Cataract research using electronic health records. *BMC ophthalmology*, 11:32, nov 2011. ISSN 1471-2415. doi: 10.1186/1471-2415-11-32. URL https://pubmed.ncbi.nlm.nih.gov/22078460https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3226674/.

[64] Marylyn D Ritchie, Shefali S Verma, Molly A Hall, et al. Electronic medical records and genomics (eMERGE) network exploration in cataract: several new potential susceptibility loci. *Molecular vision*, 20:1281–1295, sep 2014. ISSN 1090-0535. URL https://pubmed.ncbi.nlm.nih.gov/25352737https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4168835/.

[65] J T Delaney, A H Ramirez, E Bowton, et al. Predicting clopidogrel response using DNA samples linked to an electronic health record. *Clinical pharmacology and therapeutics*, 91(2):257–263, feb 2012. ISSN 1532-6535. doi: 10.1038/clpt.2011.221. URL https://pubmed.ncbi.nlm.nih.gov/22190063https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3621954/.

[66] Courtney E Walters Jr, Rachana Nitin, Katherine Margulis, et al. Automated Phenotyping Tool for Identifying Developmental Language Disorder Cases in Health Systems Data (APT-DLD): A New Research Algorithm for Deployment in Large-Scale Electronic Health Record Systems. *Journal of speech, language, and hearing research : JSLHR*, 63(9):3019–3035, sep 2020. ISSN 1558-9102. doi: 10.1044/2020_JSLHR-19-00397. URL https://pubmed.ncbi.nlm.nih.gov/32791019https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7890229/.

[67] Selen Bozkurt, Jung In Park, Kathleen Mary Kan, et al. An Automated Feature Engineering for Digital Rectal Examination Documentation using Natural Language Processing. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2018: 288–294, dec 2018. ISSN 1942-597X. URL https://pubmed.ncbi.nlm.nih.gov/30815067https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6371344/.

[68] Selen Bozkurt, Kathleen M Kan, Michelle K Ferrari, et al. Is it possible to automatically assess pretreatment digital rectal examination documentation using natural language processing? A single-centre retrospective study. *BMJ open*, 9(7): e027182–e027182, jul 2019. ISSN 2044-6055. doi: 10.1136/bmjopen-2018-027182. URL https://pubmed.ncbi.nlm.nih.gov/31324681https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6661600/.

[69] Casey Lynnette Overby, Chunhua Weng, Krystl Haerian, et al. Evaluation considerations for EHR-based phenotyping algorithms: A case study for drug-induced liver injury. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2013:130–134, mar 2013. ISSN 2153-4063. URL https://pubmed.ncbi.nlm.nih.gov/24303321https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3814479/.

[70] Casey Lynnette Overby, Jyotishman Pathak, Omri Gottesman, et al. A collaborative approach to developing an electronic health record phenotyping algorithm for drug-induced liver injury. *Journal of the American Medical Informatics Association : JAMIA*, 20(e2):e243–e252, dec 2013. ISSN 1527-974X. doi: 10.1136/amiajnl-2013-001930. URL https://pubmed.ncbi.nlm.nih.gov/23837993https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3861914/.

[71] Maya S Safarova, Hongfang Liu, and Iftikhar J Kullo. Rapid identification of familial hypercholesterolemia from electronic health records: The SEARCH study. *Journal of clinical lipidology*, 10(5):1230–1239, 2016. ISSN 1933-2874. doi: 10.1016/j.jacl.2016.08.001. URL https://pubmed.ncbi.nlm.nih.gov/27678441.

[72] A. Muthalagu, J. A. Pacheco, S. Aufox, et al. A Rigorous Algorithm To Detect And Clean Inaccurate Adult Height Records Within EHR Systems. *Applied Clinical Informatics*, 05(01):118–126, 2014. doi: 10.4338/aci-2013-09-ra-0074.

[73] D R Crosslin, D S Carrell, A Burt, et al. Genetic variation in the HLA region is associated with susceptibility to herpes zoster. *Genes and immunity*, 16(1):1–7, 2015. ISSN 1476-5470. doi: 10.1038/gene.2014.51. URL https://pubmed.ncbi.nlm.nih.gov/25297839https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4308645/.

[74] Qiping Feng, Lan Jiang, Richard L Berg, et al. A common CNR1 (cannabinoid receptor 1) haplotype attenuates the decrease in HDL cholesterol that typically accompanies weight gain. *PloS one*, 5(12):e15779–e15779, dec 2010. ISSN 1932-6203. doi: 10.1371/journal.pone.0015779. URL https://pubmed.ncbi.nlm.nih.gov/21209828https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3013130/.

[75] Stephen D Turner, Richard L Berg, James G Linneman, et al. Knowledge-driven multi-locus analysis reveals gene-gene interactions influencing HDL cholesterol level in two independent EMR-linked biobanks. *PloS one*, 6(5):e19586–e19586, may 2011. ISSN 1932-6203. doi: 10.1371/journal.pone.0019586. URL https://pubmed.ncbi.nlm.nih.gov/21589926https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3092760/.

[76] Joshua C Denny, Dana C Crawford, Marylyn D Ritchie, et al. Variants near FOXE1 are associated with hypothyroidism and other thyroid conditions: using electronic medical records for genome- and phenome-wide studies. *American journal of human genetics*, 89(4):529–542, oct 2011. ISSN 1537-6605. doi: 10.1016/j.ajhg.2011.09.008. URL https://pubmed.ncbi.nlm.nih.gov/21981779https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3188836/.

[77] Jennifer R Malinowski, Joshua C Denny, Suzette J Bielinski, et al. Genetic variants associated with serum thyroid stimulating hormone (TSH) levels in European Americans and African Americans from the eMERGE Network. *PloS one*, 9(12):

e111301–e111301, dec 2014. ISSN 1932-6203. doi: 10.1371/journal.pone.0111301. URL https://pubmed.ncbi.nlm.nih.gov/25436638https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4249871/.

[78] Laura J Rasmussen-Torvik, Jennifer A Pacheco, Russell A Wilke, et al. High density GWAS for LDL cholesterol in African Americans using electronic medical records reveals a strong protective variant in APOE. *Clinical and translational science*, 5(5): 394–399, oct 2012. ISSN 1752-8062. doi: 10.1111/j.1752-8062.2012.00446.x. URL https://pubmed.ncbi.nlm.nih.gov/23067351https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3521536/.

[79] Karishma Desai, Ian Carroll, Steven M Asch, et al. Utilization and effectiveness of multimodal discharge analgesia for postoperative pain management. *The Journal of surgical research*, 228:160–169, aug 2018. ISSN 1095-8673. doi: 10.1016/j.jss.2018.03.029. URL https://pubmed.ncbi.nlm.nih.gov/29907207https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6476628/.

[80] Iftikhar J Kullo, Jin Fan, Jyotishman Pathak, et al. Leveraging informatics for genetic studies: use of the electronic medical record to enable a genome-wide association study of peripheral arterial disease. *Journal of the American Medical Informatics Association : JAMIA*, 17(5):568–574, 2010. ISSN 1527-974X. doi: 10.1136/jamia.2010.004366. URL https://pubmed.ncbi.nlm.nih.gov/20819866https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2995686/.

[81] Iftikhar J Kullo, Keyue Ding, Hayan Jouni, et al. A genome-wide association study of red blood cell traits using the electronic medical record. *PloS one*, 5(9):e13011, sep 2010. ISSN 1932-6203. doi: 10.1371/journal.pone.0013011. URL https://pubmed.ncbi.nlm.nih.gov/20927387https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2946914/.

[82] Logan Dumitrescu, Marylyn D Ritchie, Joshua C Denny, et al. Genome-wide study of resistant hypertension identified from electronic health records. *PloS one*, 12(2): e0171745–e0171745, feb 2017. ISSN 1932-6203. doi: 10.1371/journal.pone.0171745. URL https://pubmed.ncbi.nlm.nih.gov/28222112https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5319785/.

[83] Daniel E Michalik, Bradley W Taylor, and Julie A Panepinto. Identification and Validation of a Sickle Cell Disease Cohort Within Electronic Health Records. *Academic pediatrics*, 17(3):283–287, apr 2017. ISSN 1876-2867. doi: 10.1016/j.acap.2016.12.005. URL https://pubmed.ncbi.nlm.nih.gov/27979750.

[84] Wei-Qi Wei, Xiaohui Li, Qiping Feng, et al. LPA Variants Are Associated With Residual Cardiovascular Risk in Patients Receiving Statins. *Circulation*, 138(17): 1839–1849, oct 2018. ISSN 1524-4539. doi: 10.1161/CIRCULATIONAHA.117.031356. URL

https://pubmed.ncbi.nlm.nih.gov/29703846https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6202211/.

[85] Wei-Qi Wei, Qiping Feng, Peter Weeke, et al. Creation and Validation of an EMR-based Algorithm for Identifying Major Adverse Cardiac Events while on Statins. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2014:112–119, apr 2014. ISSN 2153-4063. URL https://pubmed.ncbi.nlm.nih.gov/25717410https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4333709/.

[86] Seth E Karol, Wenjian Yang, Sara L Van Driest, et al. Genetics of glucocorticoid-associated osteonecrosis in children with acute lymphoblastic leukemia. *Blood*, 126(15):1770–1776, oct 2015. ISSN 1528-0020. doi: 10.1182/blood-2015-05-643601. URL https://pubmed.ncbi.nlm.nih.gov/26265699https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4600016/.

[87] April Barnado, Carolyn Casey, Robert J Carroll, et al. Developing Electronic Health Record Algorithms That Accurately Identify Patients With Systemic Lupus Erythematosus. *Arthritis care & research*, 69(5):687–693, may 2017. ISSN 2151-4658. doi: 10.1002/acr.22989. URL https://pubmed.ncbi.nlm.nih.gov/27390187https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5219863/.

[88] Abel N Kho, M Geoffrey Hayes, Laura Rasmussen-Torvik, et al. Use of diverse electronic medical record systems to identify genetic risk for type 2 diabetes within a genome-wide association study. *Journal of the American Medical Informatics Association : JAMIA*, 19(2):212–218, 2012. ISSN 1527-974X. doi: 10.1136/amiajnl-2011-000439. URL https://pubmed.ncbi.nlm.nih.gov/22101970https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3277617/.

[89] Wei-Qi Wei, Cynthia L Leibson, Jeanine E Ransom, et al. Impact of data fragmentation across healthcare centers on the accuracy of a high-throughput clinical phenotyping algorithm for specifying subjects with type 2 diabetes mellitus. *Journal of the American Medical Informatics Association : JAMIA*, 19(2):219–224, 2012. ISSN 1527-974X. doi: 10.1136/amiajnl-2011-000597. URL https://pubmed.ncbi.nlm.nih.gov/22249968https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3277630/.

[90] George Hripcsak, Ning Shang, Peggy L Peissig, et al. Facilitating phenotype transfer using a common data model. *Journal of biomedical informatics*, 96:103253, aug 2019. ISSN 1532-0480. doi: 10.1016/j.jbi.2019.103253. URL https://pubmed.ncbi.nlm.nih.gov/31325501https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6697565/.

[91] Imon Banerjee, Kevin Li, Martin Seneviratne, et al. Weakly supervised natural language processing for assessing patient-centered outcome following prostate cancer

treatment. *JAMIA open*, 2(1):150–159, apr 2019. ISSN 2574-2531. doi: 10.1093/jamiaopen/ooy057. URL https://pubmed.ncbi.nlm.nih.gov/31032481https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6482003/.

[92] Andrea H Ramirez, Yaping Shi, Jonathan S Schildcrout, et al. Predicting warfarin dosage in European-Americans and African-Americans using DNA samples linked to an electronic health record. *Pharmacogenomics*, 13(4):407–418, mar 2012. ISSN 1744-8042. doi: 10.2217/pgs.11.164. URL https://pubmed.ncbi.nlm.nih.gov/22329724https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3361510/.

[93] David R Crosslin, Andrew McDavid, Noah Weston, et al. Genetic variants associated with the white blood cell count in 13,923 subjects in the eMERGE Network. *Human genetics*, 131(4):639–652, apr 2012. ISSN 1432-1203. doi: 10.1007/s00439-011-1103-9. URL https://pubmed.ncbi.nlm.nih.gov/22037903https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3640990/.

[94] KNIME — Open for Innovation. URL https://www.knime.com/.

[95] Luke V. Rasmussen, Will K. Thompson, Jennifer A. Pacheco, et al. Design patterns for the development of electronic health record-driven phenotype extraction algorithms. *Journal of Biomedical Informatics*, 51:280–286, 2014. ISSN 15320464. doi: 10.1016/j.jbi.2014.06.007. URL http://dx.doi.org/10.1016/j.jbi.2014.06.007.

[96] Luke V Rasmussen, Richard C Kiefer, Huan Mo, et al. A Modular Architecture for Electronic Health Record-Driven Phenotyping. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2015:147–51, 2015. ISSN 2153-4063. URL http://www.ncbi.nlm.nih.gov/pubmed/26306258.

[97] CQL Documentation: Introduction, . URL https://cql.hl7.org/01-introduction.html.

[98] Emelia J. Benjamin, Paul Muntner, Alvaro Alonso, et al. Heart Disease and Stroke Statistics—2019 Update: A Report From the American Heart Association. *Circulation*, 139(10):e56–e528, mar 2019. ISSN 0009-7322. doi: 10.1161/CIR.0000000000000659. URL https://www.ahajournals.org/doi/10.1161/CIR.0000000000000659.

[99] George Hripcsak, Ning Shang, Peggy L. Peissig, et al. Facilitating phenotype transfer using a common data model. *Journal of Biomedical Informatics*, 96:103253, aug 2019. ISSN 15320464. doi: 10.1016/j.jbi.2019.103253.

[100] Tyler R. Ross, Daniel Ng, Jeffrey S. Brown, et al. The HMO Research Network Virtual Data Warehouse: A Public Data Model to Support Collaboration. *eGEMs (Generating Evidence & Methods to improve patient outcomes)*, 2(1):2, mar 2014.

ISSN 2327-9214. doi: 10.13063/2327-9214.1049. URL https://egems.academyhealth.org/article/10.13063/2327-9214.1049/.

[101] Richard Platt, Jeffrey S. Brown, Melissa Robb, et al. The FDA Sentinel Initiative — An Evolving National Resource. *New England Journal of Medicine*, 379(22): 2091–2093, nov 2018. ISSN 0028-4793. doi: 10.1056/NEJMp1809643. URL http://www.nejm.org/doi/10.1056/NEJMp1809643.

[102] Welcome to the ACT Network!, . URL https://www.actnetwork.us/national.

[103] Himanshu Sharma, Chengsheng Mao, Yizhen Zhang, et al. Developing a portable natural language processing based phenotyping system. *BMC Medical Informatics and Decision Making*, 19(S3):78, apr 2019. ISSN 1472-6947. doi: 10.1186/s12911-019-0786-z. URL http://arxiv.org/abs/1807.06638https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-019-0786-z.

[104] Dingcheng Li, Cory M Endle, Sahana Murthy, et al. Modeling and executing electronic health records driven phenotyping algorithms using the NQF Quality Data Model and JBoss® Drools Engine. *AMIA Annual Symposium proceedings*, 2012: 532–41, 2012. ISSN 1942-597X. URL http://www.ncbi.nlm.nih.gov/pubmed/23304325.

[105] Fast Healthcare Interoperability Resources (FHIR) Specification, . URL http://hl7.org/fhir/.

[106] Home — Measure Authoring Tool. URL https://www.emeasuretool.cms.gov/.

[107] CDS Authoring Tool — CDS Connect. URL https://cds.ahrq.gov/cdsconnect/authoring.

[108] Abhinav Goyal, Catherine R. Norton, Tracy N. Thomas, et al. Predictors of Incident Heart Failure in a Large Insured Population. *Circulation: Heart Failure*, 3(6): 698–705, nov 2010. ISSN 1941-3289. doi: 10.1161/CIRCHEARTFAILURE.110.938175. URL https://pubmed.ncbi.nlm.nih.gov/20798277/https://www.ahajournals.org/doi/10.1161/CIRCHEARTFAILURE.110.938175.

[109] Gianluigi Savarese, Ola Vedin, Domenico D'Amario, et al. Prevalence and Prognostic Implications of Longitudinal Ejection Fraction Change in Heart Failure. *JACC: Heart Failure*, 7(4):306–317, 2019. ISSN 22131779. doi: 10.1016/j.jchf.2018.11.019.

[110] Geoffrey H. Tison, Alanna M. Chamberlain, Mark J. Pletcher, et al. Identifying heart failure using EMR-based algorithms. *International Journal of Medical Informatics*, 120:1–7, 2018. ISSN 18728243. doi: 10.1016/j.ijmedinf.2018.09.016. URL https://doi.org/10.1016/j.ijmedinf.2018.09.016.

[111] Clinical Quality Language Community Projects Github, . URL https://github.com/cqframework/clinical{_}quality{_}language/wiki/CommunityProjects.

[112] PhEMA Heart Failure Use Case. Github. URL https://github.com/PheMA/heart-failure-use-case.

[113] Health Level Seven Reference Implementation Model, Version 3 – ActCode – Classes — NCBO BioPortal, . URL https://bioportal.bioontology.org/ontologies/HL7?p=classes{%}2526conceptid=C1553812.

[114] Encounter - FHIR v4.0.1, . URL https://www.hl7.org/fhir/encounter-definitions.html{#}Encounter.class.

[115] OHDSI/circe-be: CIRCE is a cohort definition and syntax compiler tool for OMOP CDMv5. URL https://github.com/OHDSI/circe-be.

[116] PheMA/cql-on-omop: Runs CQL phenotype definitions against an OMOP repository. URL https://github.com/PheMA/cql-on-omop.

[117] CQL Logical Specification. URL https://cql.hl7.org/04-logicalspecification.html.

[118] Laura Wiley and Luke Rassmusen. laurakwiley/ReviewR: ReviewR, nov 2018. URL https://doi.org/10.5281/zenodo.1488535.

[119] Jacob Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46, apr 1960. ISSN 0013-1644. doi: 10.1177/001316446002000104. URL http://journals.sagepub.com/doi/10.1177/001316446002000104.

[120] DBCG/cql_engine: Clinical Quality Language Evaluation Engine, . URL https://github.com/DBCG/cql{_}engine.

[121] HAPI FHIR - The Open Source FHIR API for Java. URL https://hapifhir.io/.

[122] CMS 2008-2010 Data Entrepreneurs' Synthetic Public Use File (DE-SynPUF) — CMS. URL https://www.cms.gov/Research-Statistics-Data-and-Systems/Downloadable-Public-Use-Files/SynPUFs/DE{_}Syn{_}PUF.

[123] OHDSI/ETL-CMS: Workproducts to ETL CMS datasets into OMOP Common Data Model. URL https://github.com/OHDSI/ETL-CMS.

[124] Mohammed Khalilia, Myung Choi, Amelia Henderson, et al. Clinical Predictive Modeling Development and Deployment through FHIR Web Services. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2015:717–726, 2015. ISSN 1942597X.

[125] Peggy L. Peissig, Luke V. Rasmussen, Richard L. Berg, et al. Importance of multi-modal approaches to effectively identify cataract cases from electronic health records. *Journal of the American Medical Informatics Association*, 19(2):225–234, 2012. ISSN 10675027. doi: 10.1136/amiajnl-2011-000456. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3277618/pdf/amiajnl-2011-000456.pdf.

[126] Shawn N. Murphy, Griffin Weber, Michael Mendis, et al. Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). *Journal of the American Medical Informatics Association*, 17(2):124–130, 2010. ISSN 10675027. doi: 10.1136/jamia.2009.000893.

[127] Christopher A. Longhurst, Robert A. Harrington, and Nigam H. Shah. A 'green button' for using aggregate patient data at the point of care. *Health Affairs*, 33(7): 1229–1235, 2014. ISSN 15445208. doi: 10.1377/hlthaff.2014.0099.

[128] Alison Callahan, Vladimir Polony, José D Posada, et al. ACE: the Advanced Cohort Engine for searching longitudinal patient records. *Journal of the American Medical Informatics Association*, mar 2021. ISSN 1067-5027. doi: 10.1093/jamia/ocab027. URL https://academic.oup.com/jamia/advance-article/doi/10.1093/jamia/ocab027/6169466.

[129] Shiqiang Tao, Licong Cui, Xi Wu, and Guo Qiang Zhang. Facilitating Cohort Discovery by Enhancing Ontology Exploration, Query Management and Query Sharing for Large Clinical Data Repositories. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2017:1685–1694, 2017. ISSN 1942597X. URL /pmc/articles/PMC5977665//pmc/articles/PMC5977665/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC5977665/.

[130] John F. Hurdle, Stephen C. Haroldsen, Andrew Hammer, et al. Identifying clinical/translational research cohorts: Ascertainment via querying an integrated multi-source database. *Journal of the American Medical Informatics Association*, 20 (1):164–171, jan 2013. ISSN 10675027. doi: 10.1136/amiajnl-2012-001050. URL https://academic.oup.com/jamia/article-lookup/doi/10.1136/amiajnl-2012-001050.

[131] Nicholas J Dobbins, Clifford H Spital, Robert A Black, et al. Leaf: an open-source, model-agnostic, data-driven web application for cohort discovery and translational biomedical research. *Journal of the American Medical Informatics Association*, 27(1): 109–118, jan 2020. ISSN 1527-974X. doi: 10.1093/jamia/ocz165. URL https://academic.oup.com/jamia/article/27/1/109/5583724.

[132] Henry J. Lowe, Todd A. Ferris, Penni M. Hernandez, and Susan C. Weber. STRIDE–An integrated standards-based translational research informatics platform. *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, 2009:391–395, 2009. ISSN 1942597X. doi: 10.1161/CIR.0000000000000152. URL /pmc/articles/PMC2815452//pmc/articles/PMC2815452/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC2815452/.

[133] Murphy, Barnett, and Chueh. Visual query tool for finding patient cohorts from a clinical data warehouse of the partners HealthCare system. *Proceedings. AMIA Symposium*, page 1174, 2000. ISSN 1531-605X. URL http://www.ncbi.nlm.nih.gov/pubmed/11080028http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2243863.

[134] Licong Cui, Ningzhou Zeng, Matthew Kim, et al. X-search: An open access interface for cross-cohort exploration of the National Sleep Research Resource 08 Information and Computing Sciences 0806 Information Systems. *BMC Medical Informatics and Decision Making*, 18(1):1–10, nov 2018. ISSN 14726947. doi: 10.1186/s12911-018-0682-y. URL https://link.springer.com/articles/10.1186/s12911-018-0682-yhttps://link.springer.com/article/10.1186/s12911-018-0682-y.

[135] Xingmin Aaron Zhang, Amy Yates, Nicole Vasilevsky, et al. Semantic integration of clinical laboratory tests from electronic health records for deep phenotyping and biomarker discovery. *npj Digital Medicine*, 2(1), 2019. ISSN 2398-6352. doi: 10.1038/s41746-019-0110-4. URL http://dx.doi.org/10.1038/s41746-019-0110-4.

[136] Neil S. Zheng, Qiping Feng, V. Eric Kerchberger, et al. PheMap: A multi-resource knowledge base for high-throughput phenotyping within electronic health records. *Journal of the American Medical Informatics Association*, 27(11):1675–1687, 2020. ISSN 1527974X. doi: 10.1093/jamia/ocaa104.

[137] Adam Wilcox, David Vawdrey, Chunhua Weng, et al. Research Data Explorer: Lessons Learned in Design and Development of Context-based Cohort Definition and Selection. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2015(7):194–8, 2015. ISSN 2153-4063. URL http://www.ncbi.nlm.nih.gov/pubmed/26306267.

[138] Jie Xu, Luke V Rasmussen, Pamela L Shaw, et al. Review and evaluation of electronic health records-driven phenotype algorithm authoring tools for clinical and translational research. *Journal of the American Medical Informatics Association*, page ocv070, jul 2015. ISSN 1067-5027. doi: 10.1093/jamia/ocv070. URL https://academic.oup.com/jamia/article-lookup/doi/10.1093/jamia/ocv070.

[139] P V Lee. *Automated Injection of Curated Knowledge Into Real-Time Clinical Systems CDS Architecture for the 21st Century*. PhD thesis, Arizona State University, 2018. URL https://repository.asu.edu/attachments/209644/content/Lee{_}asu{_}0010E{_}18312.pdf.

[140] Christian Maier, Lorenz A. Kapsner, Sebastian Mate, et al. Patient Cohort Identification on Time Series Data Using the OMOP Common Data Model. *Applied Clinical Informatics*, 12(1):57–64, 2021. ISSN 18690327. doi: 10.1055/s-0040-1721481.

[141] Martin Chapman, Luke V Rasmussen, Jennifer A Pacheco, and Vasa Curcin. Phenoflow: Portable Workflow-based Phenotype Definitions. *medRxiv*, (Cdm): 2020.07.01.20144196, 2020. doi: 10.1101/2020.07.01.20144196. URL https://www.medrxiv.org/content/10.1101/2020.07.01.20144196v1?{%}253fcollection=.

[142] P Amstutz, B Chapman, J Chilton, et al. Common Workflow Language, v1.0 Common Workflow Language (CWL) Command Line Tool Description, v1.0. 2016.

doi: 10.6084/m9.figshare.3115156.v2. URL https://w3id.org/cwl/v1.0/{%}0Ahttps://w3id.org/cwl/.

[143] Spiros Denaxas, Arturo Gonzalez-Izquierdo, Kenan Direk, et al. UK phenomics platform for developing and validating EHR phenotypes: CALIBER, feb 2019. URL https://doi.org/10.1101/539403.

[144] Enid Montague, Jie Xu, Luke V Rasmussen, et al. Usability of a phenotype builder prototype and lessons learned for the design of phenotyping tools [Poster]. In *AMIA Annual Symposium Proceedings*, 2015.

[145] James F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, nov 1983. ISSN 0001-0782. doi: 10.1145/182.358434. URL https://dl.acm.org/doi/10.1145/182.358434.

[146] Mario Villamizar, Oscar Garcés, Harold Castro, et al. Evaluating the Monolithic and the Microservice Architecture Pattern to Deploy Web Applications in the Cloud Evaluando el Patrón de Arquitectura Monolítica y de Micro Servicios Para Desplegar Aplicaciones en la Nube. *10th Computing Colombian Conference*, pages 583–590, 2015.

[147] Justin Guinney and Julio Saez-Rodriguez. Alternative models for sharing confidential biomedical data. *Nature Biotechnology*, 36(5):391–392, 2018. ISSN 15461696. doi: 10.1038/nbt.4128. URL http://dx.doi.org/10.1038/nbt.4128.