

OMOP Common Data Model Extract, Transform & Load

OHDSI Symposium 2022

Clair Blacketer, Melanie Philofsky





Agenda Total time 9:00-10:50

Time	Agenda item
9:00 – 9:25	Introduction to OMOP CDM
9:25 – 9:50	OHDSI ETL Best Practices
9:50 – 10:00	Energy Break
10:00 – 10:25	ETL Exercise
10:25 – 10:50	CDM & Vocabulary Exercise



Leads

Clair Blacketer



Melanie Philofsky





Helpful Bookmarks

<https://ohdsi.github.io/CommonDataModel/>



OMOP Common Data Model

The Observational Medical Outcomes Partnership (OMOP) Common Data Model (CDM) is an open community data standard, designed to standardize the structure and content of observational data and to enable efficient analyses that can produce reliable evidence. A central component of the OMOP CDM is the OHDSI standardized vocabularies. The OHDSI vocabularies allow organization and standardization of medical terms to be used across the various clinical domains of the OMOP common data model and enable standardized analytics that leverage the knowledge base when constructing exposure and outcome phenotypes and other features within characterization, population-level effect estimation, and patient-level prediction studies.

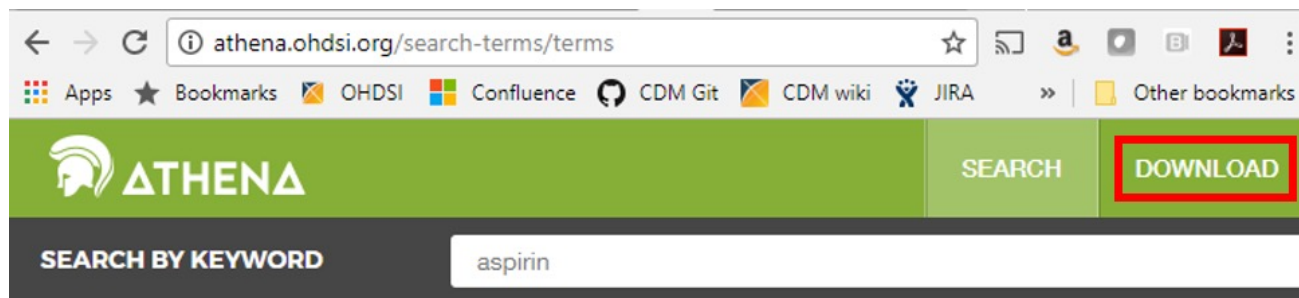
This website is meant to serve as a resource describing the specification of the available versions of the Common Data Model. This includes the structure of the model itself and the agreed upon conventions for each table and field as decided by the OHDSI Community. The vocabulary tables are part of the model and, as such, are detailed here. To download the vocabulary itself, please visit <https://athena.ohdsi.org>. For more information about the OHDSI suite of tools designed to implement best practices in characterization, population-level effect estimation and patient-level prediction, please visit <https://ohdsi.github.io/Hades/>.

Current CDM Version

The current CDM version is [CDM v5.4](#), depicted below. This CDM version was developed over the course of a year by considering requests that were sent via our [issues page](#). The list of proposed changes was then shared with the community in multiple ways: through discussions at the weekly OHDSI Community calls, discussions with the OHDSI Steering Committee, and discussions with all potentially affected workgroups. The [final changes](#) were then delivered to the Community through a new R package designed to dynamically generate the DDLs and documentation for all supported SQL dialects.

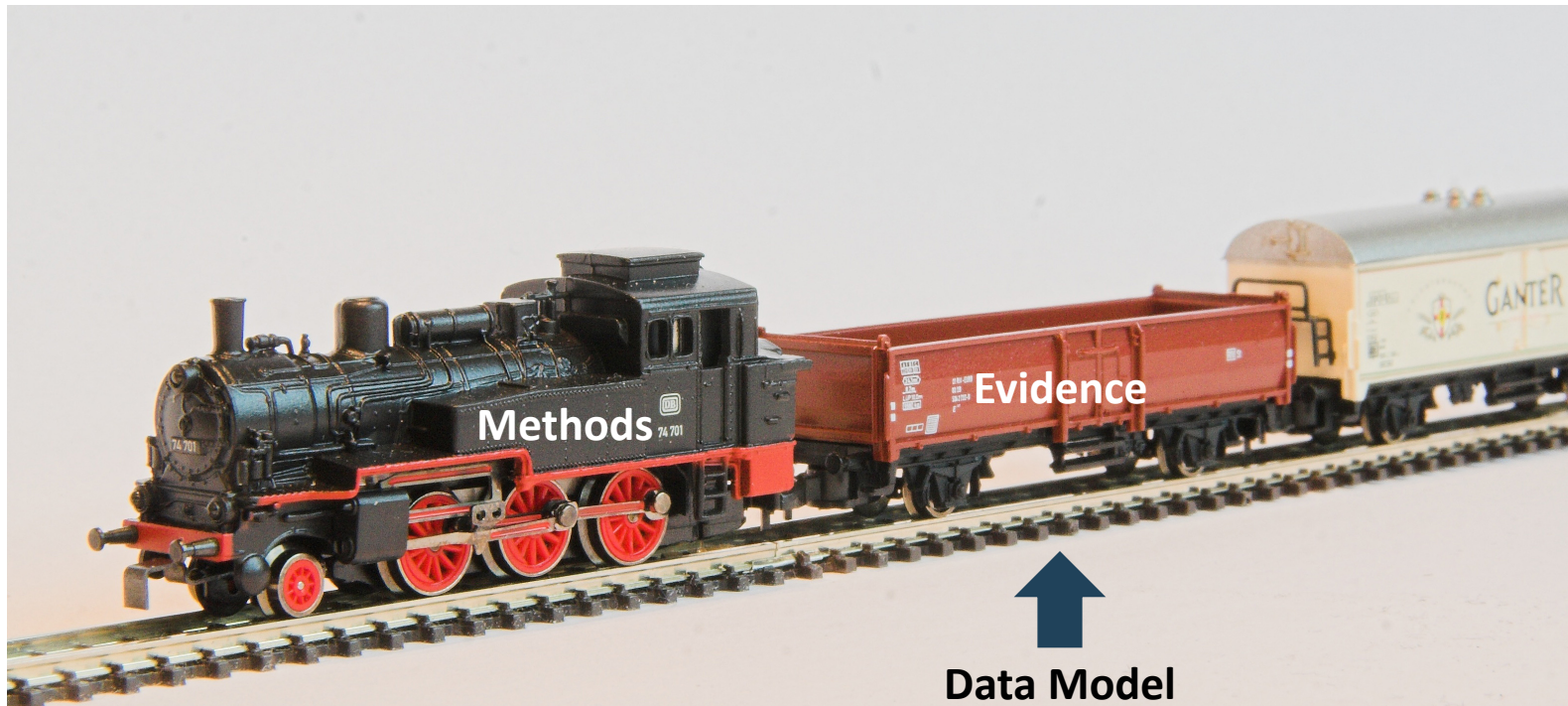
- [Link to DDLs for CDM v5.4](#)
- [Link to README for instructions on how to use the R package](#)

<https://athena.ohdsi.org>





Why a Common Data Model





Why a Common Data Model





Why a Common Data Model

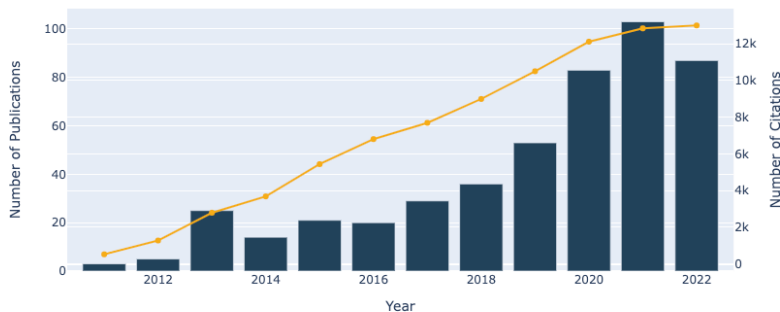


Community Dashboard Dashboards ▾

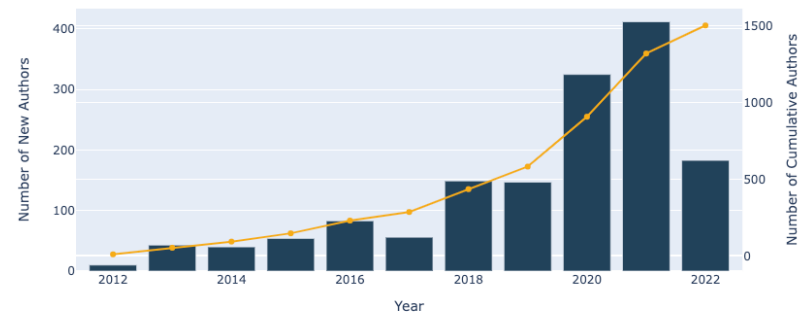
Publication Analysis

PubMed Publication Tracking highlights scholarship generated using the OMOP Common Data Model, OHDSI tools, or the OHDSI network. These publications represent scientific accomplishments across areas of data standards, methodological research, open-source development, and clinical applications. We provide the resource to search and browse the catalogue of OHDSI-related publications by date, author, title, journal, and SNOMED terms. We monitor the impact of our community using summary statistics (number of publications and citations), and the growth and diversity of our community with the number of distinct authors. Searches for new papers are performed daily, and citation counts are updated monthly.

OHDSI Publications & Cumulative Citations



New and Cumulative OHDSI Researchers





OMOP CDM

The OMOP CDM is a system of tables, vocabularies, and conventions that allow observational health data to be standardized. It is this standard approach that facilitates rapid innovation in the areas of open-source development, methods research, and evidence generation.

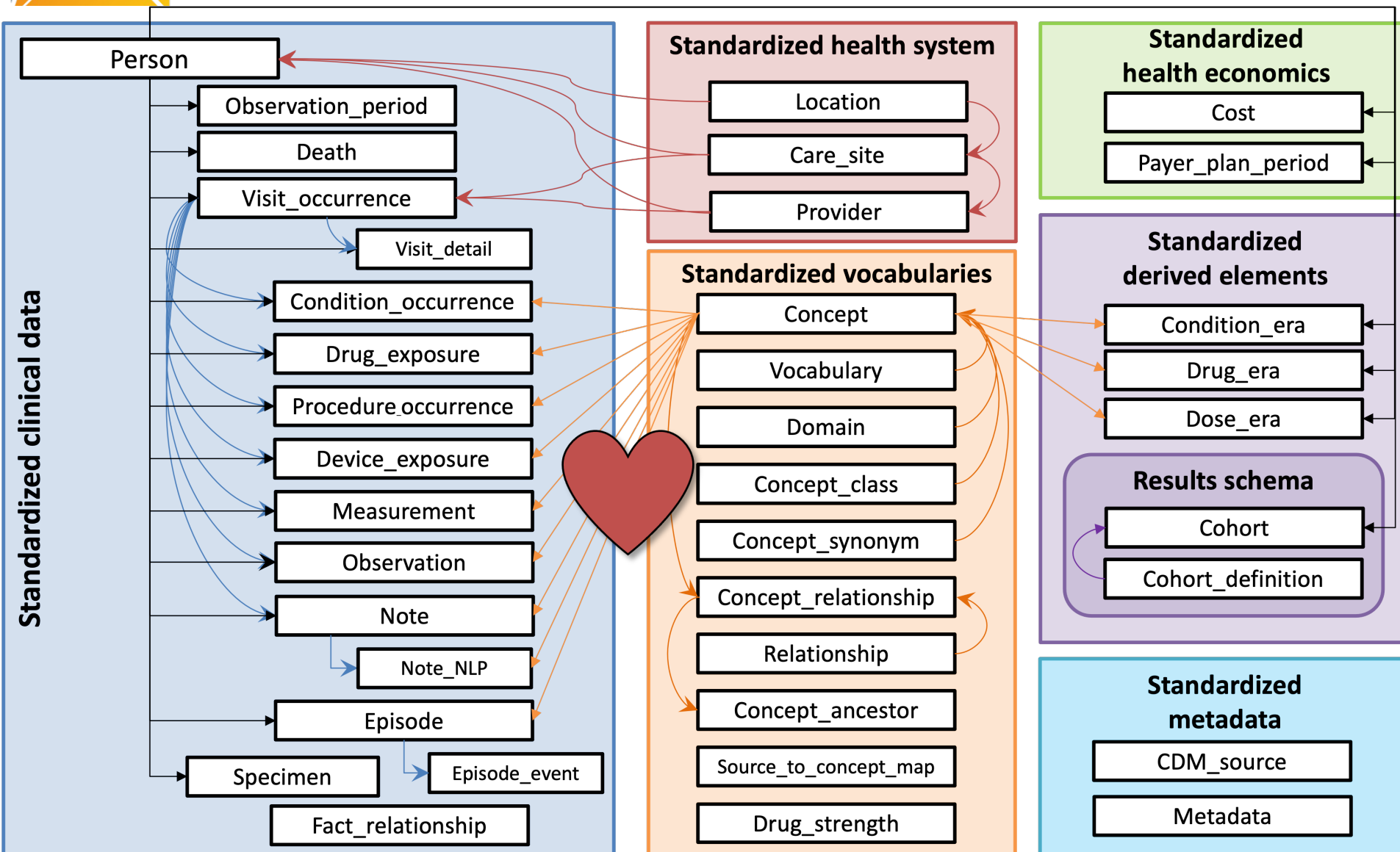


Evidence
Methods Research
Open-Source Development
Standard Analytics

CDM

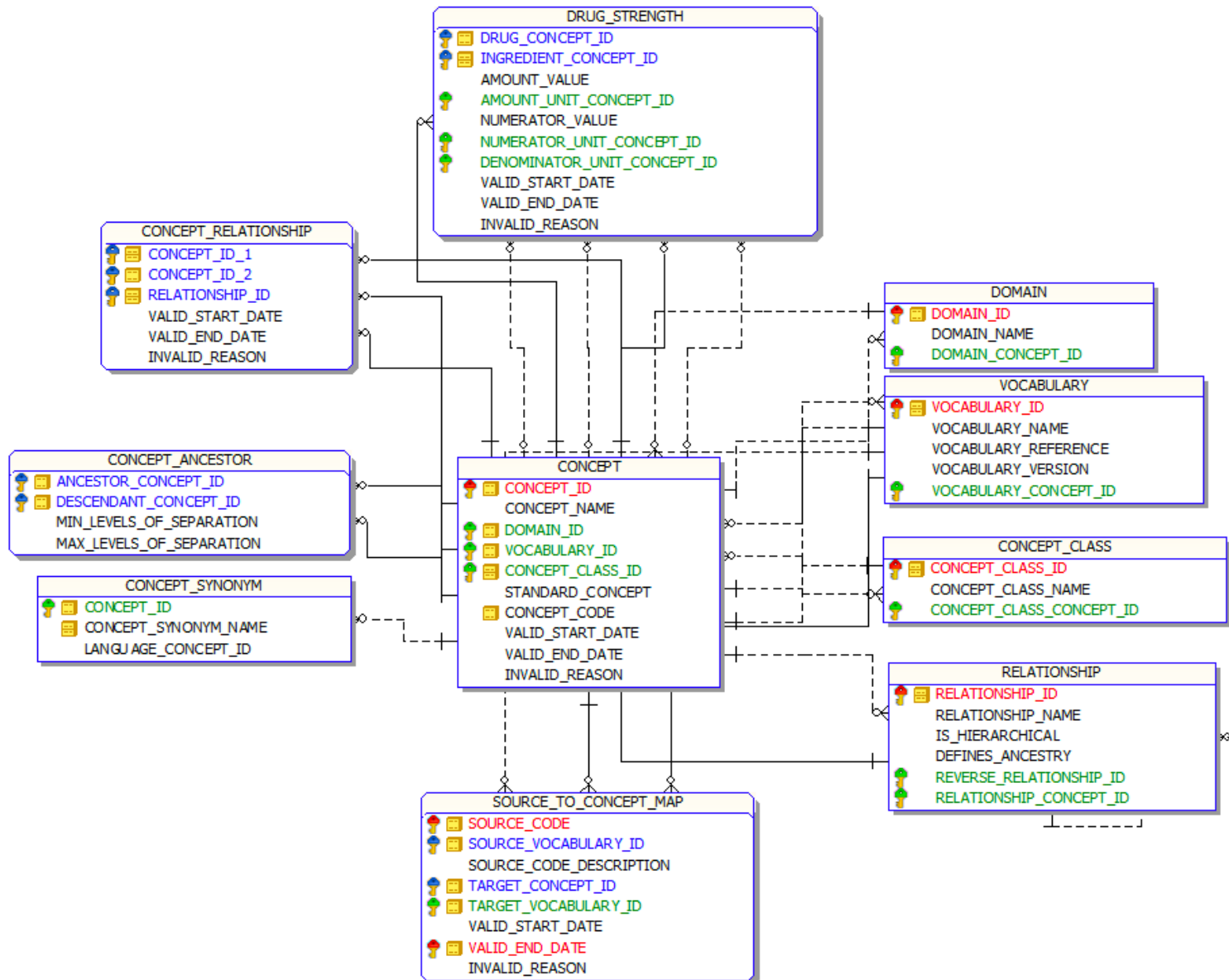


OMOP CDM & Vocabulary





OMOP Vocabulary





Different Categories of Concepts

**Non-
standard
Concepts**

Function

Unique
representation of a
source code

**Standard
Concepts**

Function

Used for standardized
analytics and by
OHDSI tools

**Classification
Concepts**

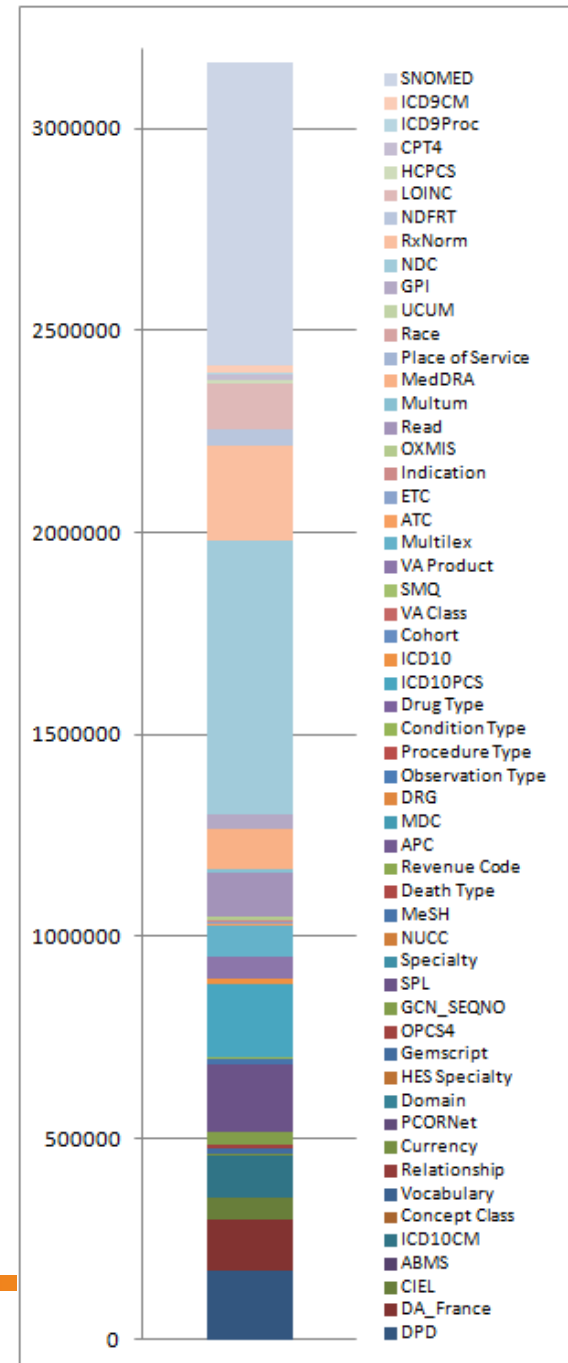
Function

Used to perform
hierarchical queries



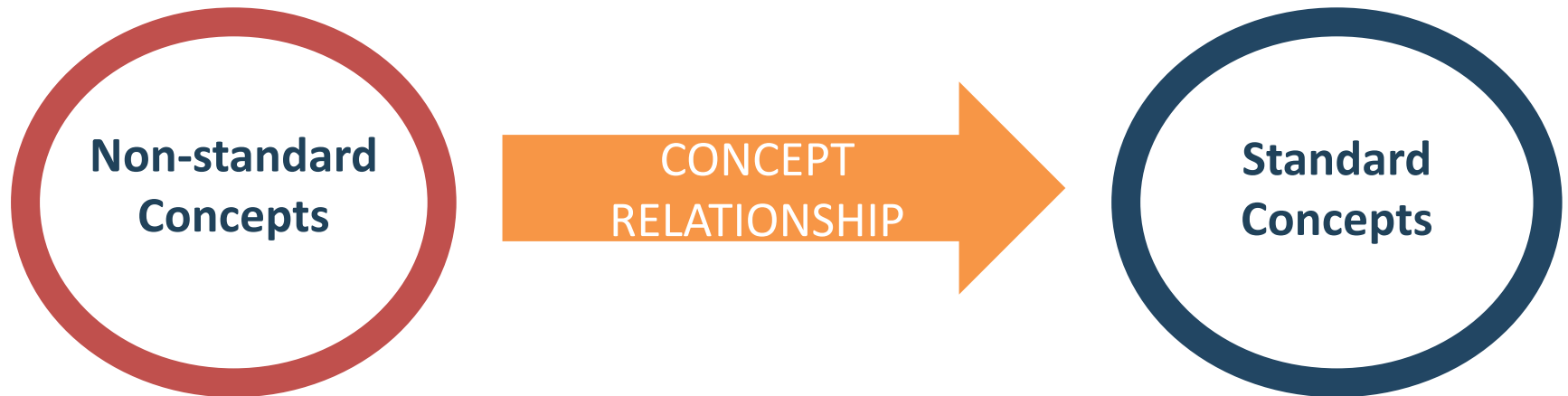
OMOP Vocabulary

- If your source data's codes are in the OMOP Vocab you can use it to translate to a standard
- For example:
 - ICD9 → SNOMED
 - NDC → RxNORM



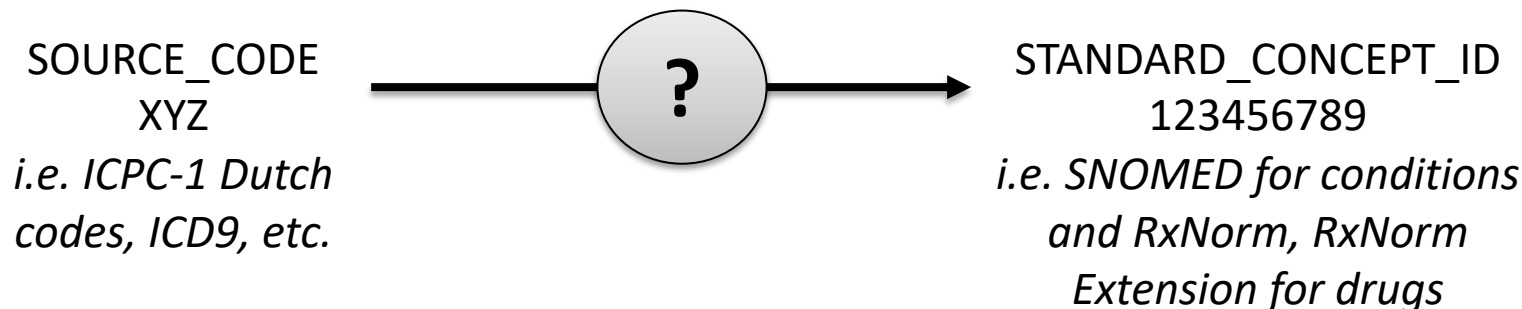


Mapping to Standard Concepts





Standardizing Terminologies



What is standardized:

1. *TABLE_CONCEPT_ID*
standard concept the source code maps to, **used for analysis**
2. *TABLE_SOURCE_CONCEPT_ID*
concept representation of the source code, **helps maintain tie to raw data**
 - a. *TABLE_SOURCE_VALUE*
original source code as given in the source table, **helps to review data quality**

Ways to get a source code to standard code:

1. OMOP Vocabulary (Concept_Relationship)
2. USAGI

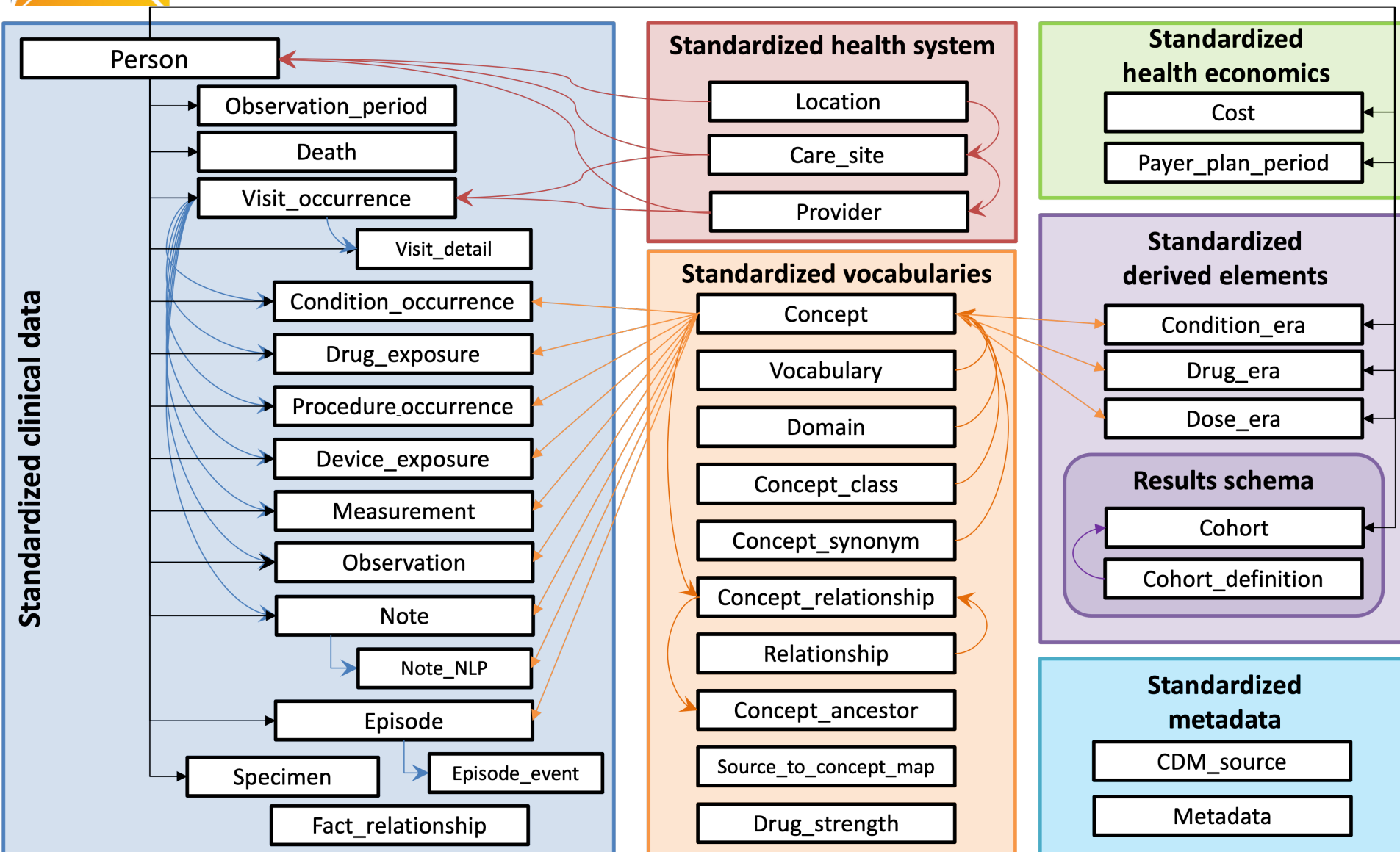


OMOP Vocab



- There are two standard queries to help us use the OMOP Vocabulary:
 - SOURCE_TO_STANDARD.sql
 - SOURCE_TO_SOURCE.sql
- <https://ohdsi.github.io/CommonDataModel/sqlScripts.html>

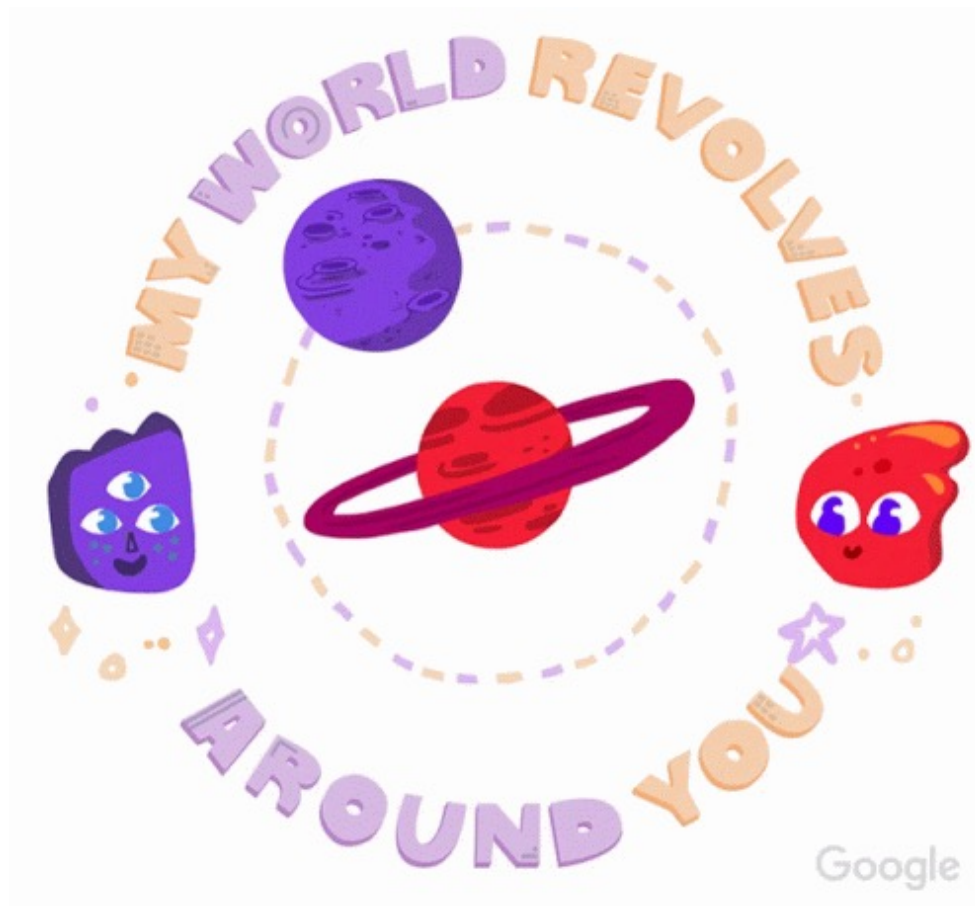
OMOP CDM





General Conventions

The OMOP CDM is a Person centric model





Table/Field conventions

PERSON

Table Description

This table serves as the central identity management for all Persons in the database. It contains records that uniquely identify each person or patient, and some demographic information.

User Guide

All records in this table are independent Persons.

ETL Conventions

All Persons in a database needs one record in this table, unless they fail data quality requirements specified in the ETL. Persons with no Events should have a record nonetheless. If more than one data source contributes Events to the database, Persons must be reconciled, if possible, across the sources to create one single record per Person. The content of the BIRTH_DATETIME must be equivalent to the content of BIRTH_DAY, BIRTH_MONTH and BIRTH_YEAR.

CDM Field	User Guide	ETL Conventions	Datatype	Required	Primary Key	Foreign Key	FK Table	FK Domain
person_id	It is assumed that every person with a different unique identifier is in fact a different person and should be treated independently.	Any person linkage that needs to occur to uniquely identify Persons ought to be done prior to writing this table. This identifier can be the original id from the source data provided if it is an integer, otherwise it can be an autogenerated number.	integer	Yes	Yes	No		



General conventions

- Required tables: person and observation_period
- Fields:
 - _id
 - _concept_id
 - _source_concept_id
 - _source_value
 - _type_concept_id
- Target concept domain determines target table

Fields

Variable names across all tables follow one convention:

Notation	Description
<code>_SOURCE_VALUE</code>	Verbatim information from the source data, typically used in ETL to map to <code>CONCEPT_ID</code> , and not to be used by any standard analytics. For example, <code>CONDITION_SOURCE_VALUE = '787.02'</code> was the ICD-9 code captured as a diagnosis from the administrative claim.
<code>_ID</code>	Unique identifiers for key entities, which can serve as foreign keys to establish relationships across entities. For example, <code>PERSON_ID</code> uniquely identifies each individual. <code>VISIT_OCCURRENCE_ID</code> uniquely identifies a <code>PERSON</code> encounter at a point of care.
<code>_CONCEPT_ID</code>	Foreign key into the Standardized Vocabularies (i.e. the <code>standard_concept</code> attribute for the corresponding term is true), which serves as the primary basis for all standardized analytics. For example, <code>CONDITION_CONCEPT_ID = 31967</code> contains the reference value for the SNOMED concept of 'Nausea'
<code>_SOURCE_CONCEPT_ID</code>	Foreign key into the Standardized Vocabularies representing the concept and terminology used in the source data, when applicable. For example, <code>CONDITION_SOURCE_CONCEPT_ID = 45431665</code> denotes the concept of 'Nausea' in the Read terminology; the analogous <code>CONDITION_CONCEPT_ID</code> might be 31967, since SNOMED-CT is the Standardized Vocabulary for most clinical diagnoses and findings.
<code>_TYPE_CONCEPT_ID</code>	Delineates the origin of the source information, standardized within the Standardized Vocabularies. For example, <code>DRUG_TYPE_CONCEPT_ID</code> can allow analysts to discriminate between 'Pharmacy dispensing' and 'Prescription written'



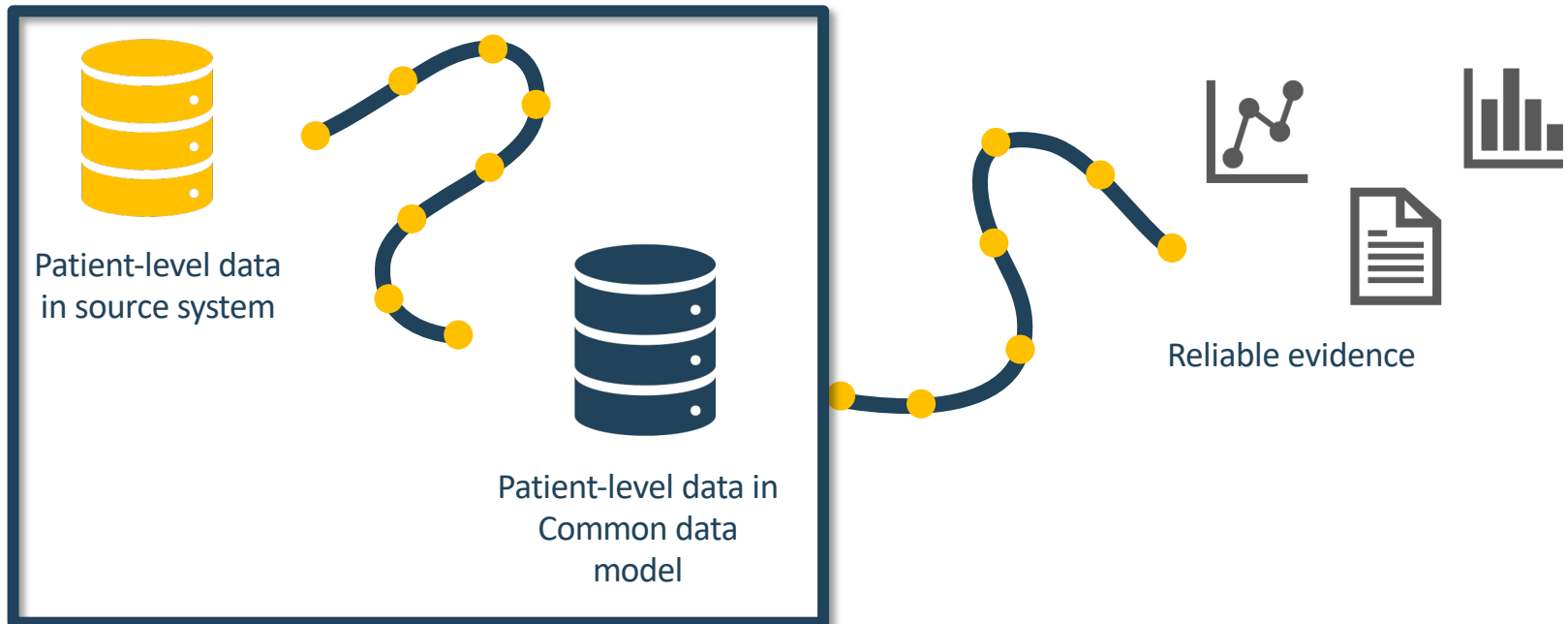
ETL Process and Tools

- Best Practices
- ETL Process
- ETL Tools
 - White Rabbit tool - review the output
 - Rabbit in a Hat tool - document the conceptual logic
 - Usagi - mapping custom source values



ETL

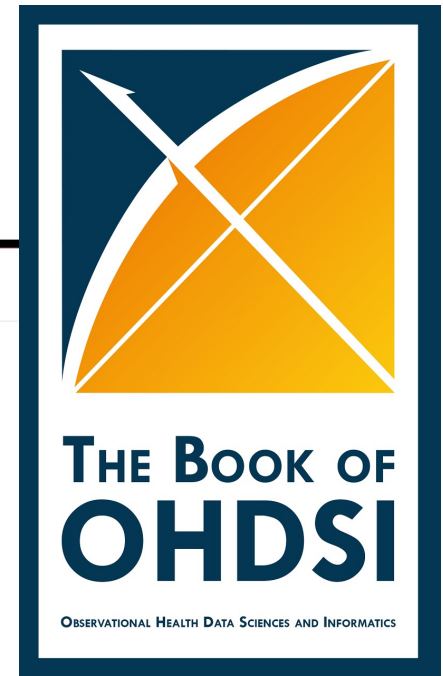
- Extract Transform Load
- In order to get from our native/raw data into the OMOP CDM we need to design and develop and ETL process



- Goal in ETLing is to standardize the format and terminology



ETL Process

A screenshot of the 'The Book of OHDSI' website. The left sidebar shows a table of contents with '6 Extract Transform Load' selected. The main content area displays the title 'Chapter 6 Extract Transform Load', the authors 'Clair Blacketer & Erica Voss', and the section '6.1 Introduction'. The introduction text explains the ETL process and lists four major steps for creating an ETL.

The Book of OHDSI

Preface

I The OHDSI Community

1 The OHDSI Community

2 Where to Begin

3 Open Science

II Uniform Data Representation

4 The Common Data Model

5 Standardized Vocabularies

6 Extract Transform Load

6.1 Introduction

6.2 Step 1: Design the ETL

6.3 Step 2: Create the Code Map...

6.4 Step 3: Implement the ETL

6.5 Step 4: Quality Control

6.6 ETL Conventions and THEMIS

6.7 CDM and ETL Maintenance

Chapter 6 Extract Transform Load

Chapter leads: Clair Blacketer & Erica Voss

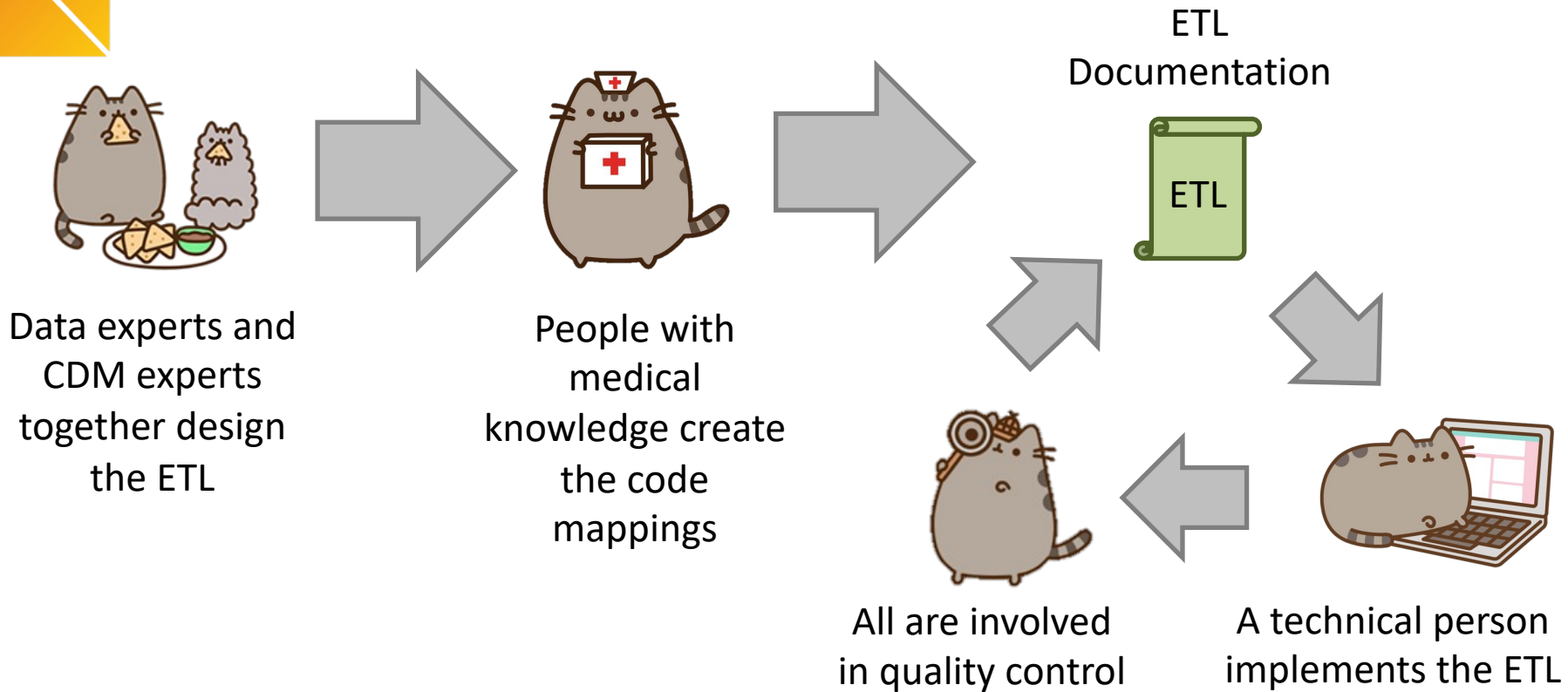
6.1 Introduction

In order to get from the native/raw data to the OMOP Common Data Model (CDM) we have to create an extract, transform, and load (ETL) process. This process should restructure the data to the CDM, and add mappings to the Standardized Vocabularies, and is typically implemented as a set of automated scripts, for example SQL scripts. It is important that this ETL process is repeatable, so that it can be rerun whenever the source data is refreshed.

Creating an ETL is usually a large undertaking. Over the years, we have developed best practices, consisting of four major steps:

1. Data experts and CDM experts together design the ETL.
2. People with medical knowledge create the code mappings.
3. A technical person implements the ETL.

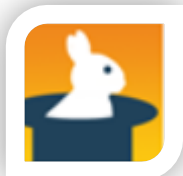
ETL Process



OHDSI Tools



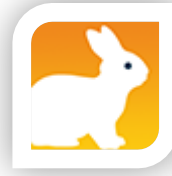
White Rabbit



Rabbit In a Hat



Usagi



White Rabbit



ACHILLES



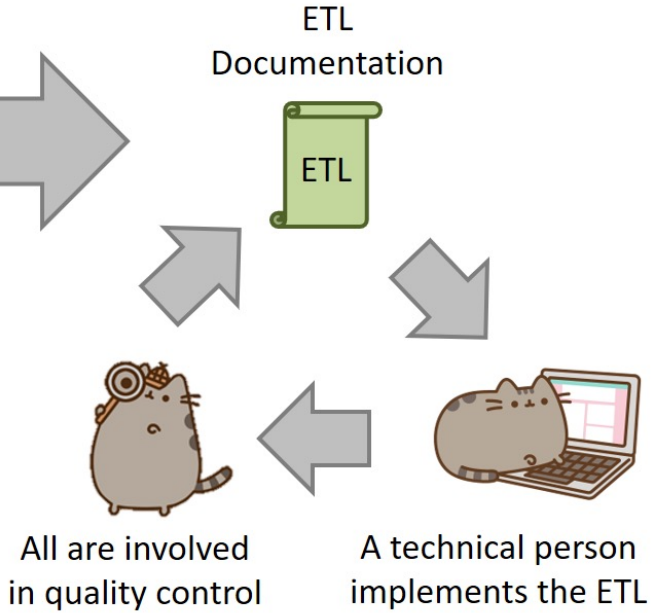
DQD



Rabbit In a Hat



OHDSI
OBSERVATIONAL HEALTH DATA SCIENCES AND INFORMATICS

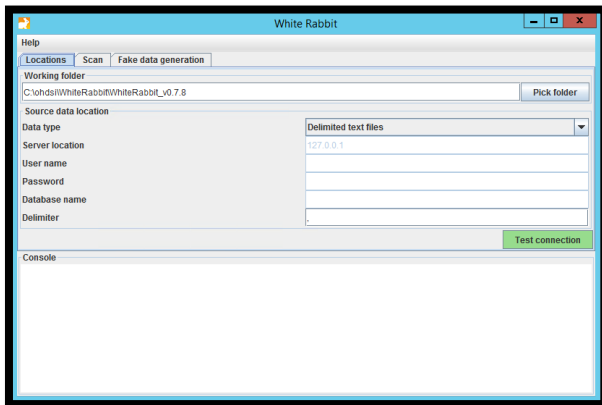




White Rabbit



- White Rabbit scans source data & creates a csv report on the source data
- The scan can be used to:
 - Learn about your source data
 - Help design the ETL
 - Used by Rabbit In a Hat





WR Output – ScanReport.xlsx



Table/Field Overview

Table	Field	Description	Type	Max length	N rows
pop	der_sex		character	1	16374539
pop	der_yob		double pre	6	16374539
pop	pat_id		character	64	16374539
pop	pat_hash_id		character	16	16374539
pop	pmtx_flag		numeric	1	16374539
pop	anon_ims_pat_id		character	11	16374539
pop	pat_region		character	2	16374539
pop	pat_state		character	2	16374539
pop	pat_zip3		character	3	16374539
pop	grp_indv_cd		character	1	16374539
pop	mh_cd		character	1	16374539
pop	enr_rel		character	2	16374539
pop	temp_col1		character	0	16374539
pop	temp_col2		character	0	16374539
pop	load_row_id		bigint	9	16374539
claims_diag_lk	person_source_valu		character	64	2992046684
claims_diag_lk	event_start_date		date	10	2992046684
claims_diag_lk	event_end_date		date	10	2992046684

Value counts

	A	B	C	D	
1	der_sex	Frequency	der_yob	Frequency	pa
2	F	50479	1991.0	2030	Li
3	M	49514	1992.0	1970	
4	U	7	1990.0	1947	
5			1989.0	1908	
6			1988.0	1873	
7			1994.0	1872	
8			1995.0	1806	
9			1993.0	1805	
10			1996.0	1716	
11			1986.0	1676	
12			1987.0	1643	
13			1985.0	1633	
14			1983.0	1588	
15			1981.0	1581	
16			1984.0	1576	
17			1970.0	1555	
18			1980.0	1553	

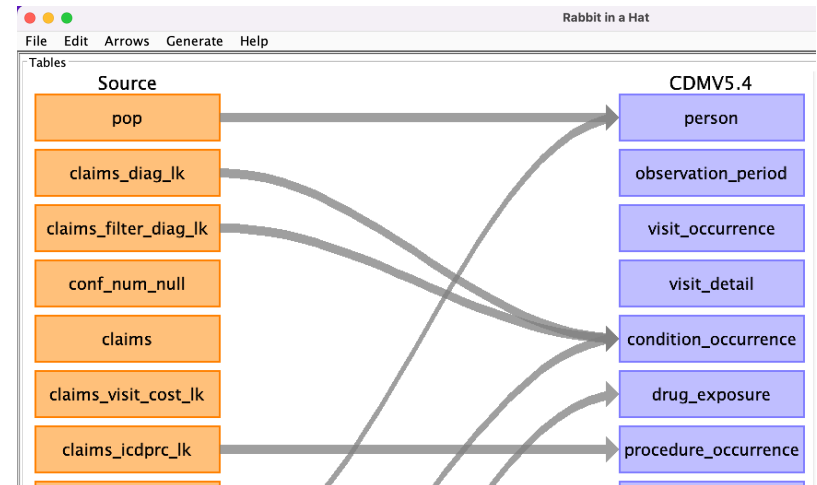
Navigation bar: pop | claims_diag_lk | claims



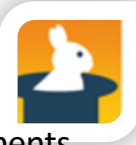
Rabbit in a Hat



- Read and display a White Rabbit scan document
- Provides a graphical interface to allow a user to connect source data to CDM tables



RiaH - Output



Word document

Table with columns: measurement_time, measurement_type_concept_id, operator_concept_id, unit_concept_id, range_low, range_high, provider_id, visit_occurrence_id, visit_detail_id, measurement_source_concept_id, unit_source_value.

Table name: observation
Reading from diagnostics
'History of'

Diagram showing Source fields (*subject_id, date_diag_875_i1, history_solitar_plasmocyt_i1) mapping to Destination fields (*person_id, *observation_concept_id, *observation_date).

Destination Field	Source Field	Logic	Comment
observation_id	subject_id		Auto-increment
person_id	subject_id		
observation_concept_id	history_solitar	Map to a custom concept 'History of solitary plasmacytoma'	
observation_date	date_diagnosis		
observation_datetime	date_diagnosis	380015486	Registered from EHR
value_as_number			
value_as_string			
value_as_concept_id			

Markdown documents

```
layout: default
title: Person
nav_order: 1
parents: CDM Synthea v1
description: "Person mapping from patients.csv"

# Person
## Reading from Synthea table patients.csv



| Destination Field | Source field | Logic | Comment field | |
|---|---|---|---|---|
| person_id | | Autogenerate | |
| gender_concept_id | gender | When gender = 'M' then set gender_concept_id to 8507; when gender = 'F' then set to 8532 | Drop any rows with missing/unknown gender. |
| year_of_birth | birthdate | Take year from birthdate | |
| month_of_birth | birthdate | Take month from birthdate | |
| day_of_birth | birthdate | Take day from birthdate | |
| birth_datetime | birthdate | With midnight as time 00:00:00 | |
| race_concept_id | race | When race = 'WHITE' then set as 8527, when race = 'LACK' then set as 8516, when race = 'ASIAN' then set as 8515, otherwise set as | |
| ethnicity_concept_id | race | ethnicity | When race = 'HISPANIC', or when hnicity in ('CENTRAL_AMERICAN', 'DOMINICAN', 'MEXICAN', 'PUERTO_RICAN', 'OUTH_AMERICAN') then set as 38803563, otherwise set as 0 | |
| location_id | | | |
| provider_id | | | |
| core_site_id | | | |
| person_source_value | id | | |
| gender_source_value | gender | | |
| gender_source_concept_id | | | |
| race_source_value | race | | |
| race_source_concept_id | | | |
| ethnicity_source_value | ethnicity | | |
| ethnicity_source_concept_id | | | |
```

Html

Tutorial-ETL

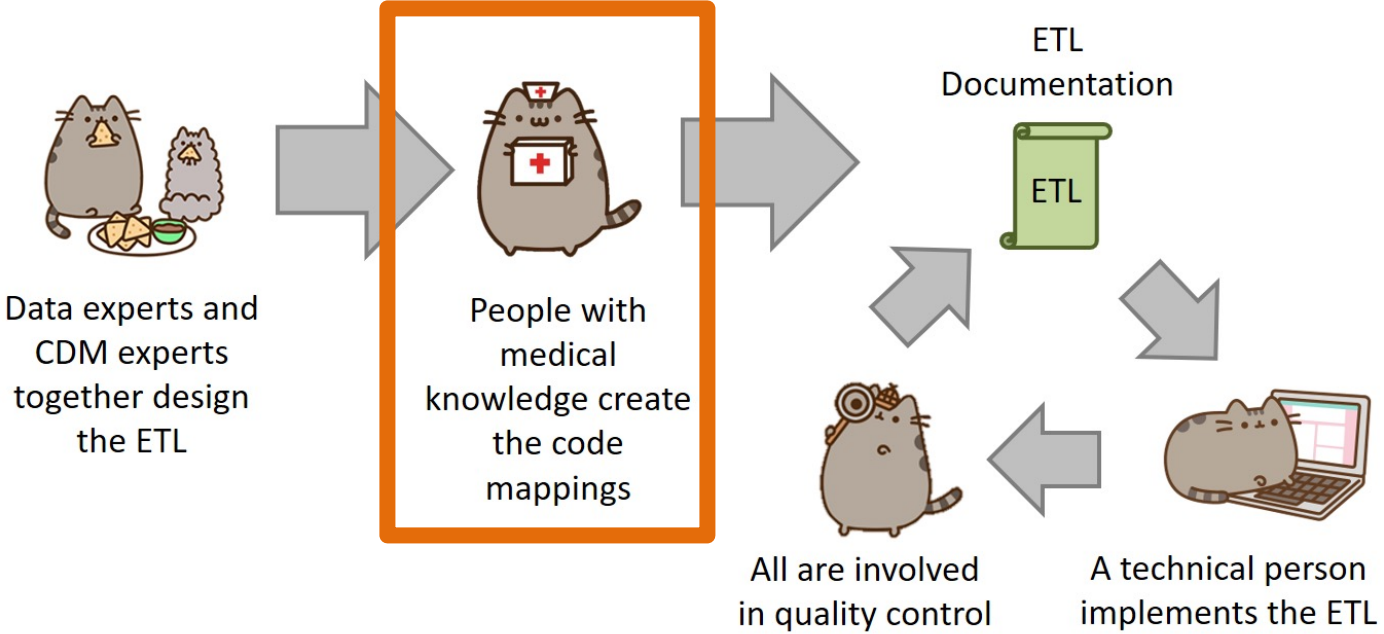
Person

Reading from Synthea table patients.csv

Destination Field	Source field	Logic	Comment field
person_id		Autogenerate	
gender_concept_id	gender	When gender = 'M' then set gender_concept_id to 8507; when gender = 'F' then set to 8532	Drop any rows with missing/unknown gender.
year_of_birth	birthdate	Take year from birthdate	
month_of_birth	birthdate	Take month from birthdate	
day_of_birth	birthdate	Take day from birthdate	
birth_datetime	birthdate	With midnight as time 00:00:00	
		When race = 'WHITE' then set as 8527, when	



OHDSI
OBSERVATIONAL HEALTH DATA SCIENCES AND INFORMATICS





Usagi



- When the Vocabulary does not contain your source terms you will need to create a map to OMOP Vocabulary Concepts
- Usagi helps you to:
 - Find best matches, automatically and/or manually
 - Automatic matching based on text similarities (itf/df)
 - Create ‘source to concept map’

The screenshot shows the Usagi application window. At the top, there is a menu bar with 'File', 'Edit', 'View', and 'Help'. Below the menu bar is a table with columns: Status, Source code, Source term, Frequency, ICPC_DES, Match score, Concept ID, Concept na., Domain, Concept cl., Vocabulary, Concept co., Standard c., Parents, Children, and Comment. The table lists several source codes and their matches, such as 'A97' (No illness) matching '4192174' (Illness) with a match score of 0.82.

Below the table, there is a section for 'Source code' with fields for Source code, Source term, Frequency, and ICPC_DESCRIPTION_DUTCH. The 'Source code' field contains 'A97' and 'No illness', 'Source term' contains '500000', and 'Frequency' contains 'Geen ziekte'.

Below that is a section for 'Target concepts' with columns: Concept ID, Concept name, Domain, Concept class, Vocabulary, Concept code, Standard concept, Parents, and Children. The table shows a match for '4192174' (Illness) with a match score of 0.82.

Below the target concepts is a 'Search' section with a 'Query' field and a 'Filters' section. The 'Query' field contains 'A97' and the 'Filters' section has checkboxes for 'Filter by user selected concepts', 'Filter by concept class', 'Filter standard concepts', 'Filter by vocabulary', 'Include source terms', and 'Filter by domain'.

Below the search section is a 'Results' section with a table showing the results of the search. The table has columns: Score, Term, Concept ID, Concept name, Domain, Concept class, Vocabulary, Concept code, Standard concept, Parents, and Children. The results show several matches, such as '0.82' (Illness) matching '4192174' (Illness) with a match score of 0.82.

At the bottom of the window, there is a 'Comment' field and an 'Approve' button. The status bar at the bottom shows 'Approved / total: 0 / 12 0.0% of total frequency' and 'Vocabulary version: v5.0.19-NOV-18'.



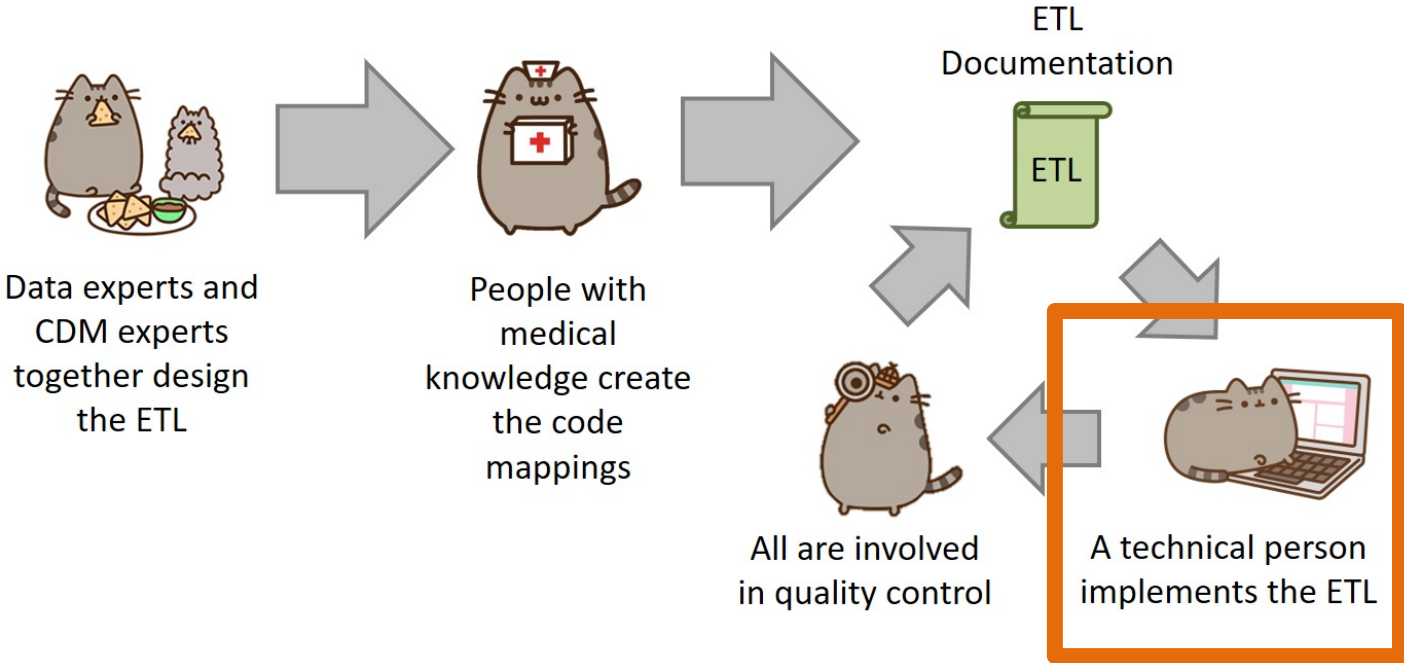
Overview - Steps



1. Get a copy of the Vocabulary from ATHENA
 2. Download Usagi
 - 3. Have Usagi build an index on the Vocabulary**
 4. Load your source codes and let Usagi process them
 5. Review and update suggested mappings with someone who has medical knowledge
 6. Export codes into the SOURCE_TO_CONCEPT_MAP
- } One-time setup



OHDSI
OBSERVATIONAL HEALTH DATA SCIENCES AND INFORMATICS





ETL Implementation



There are multiple tools available to implement your ETL

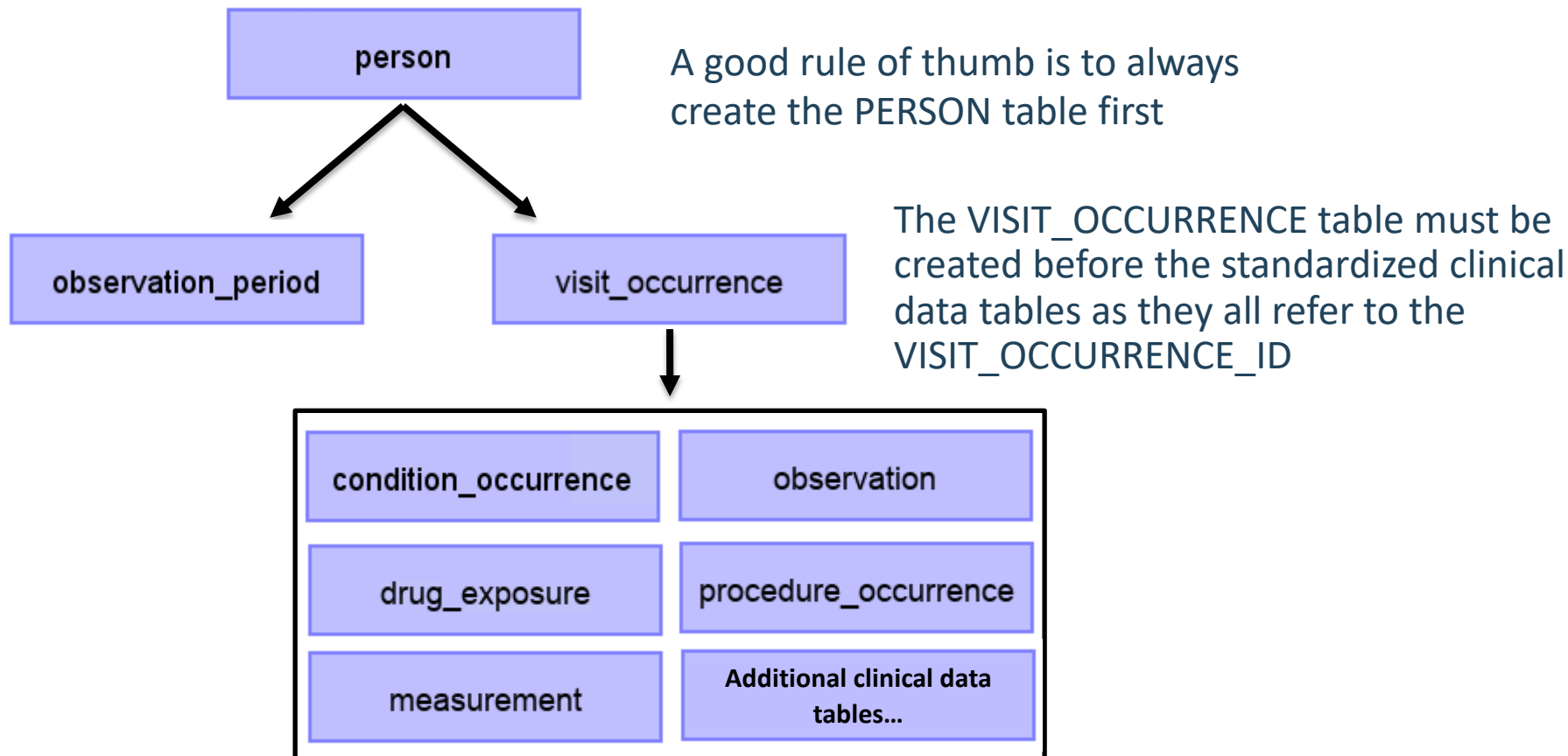


Your choice will largely depend on the size and complexity of the ETL design. And the tools available to you.



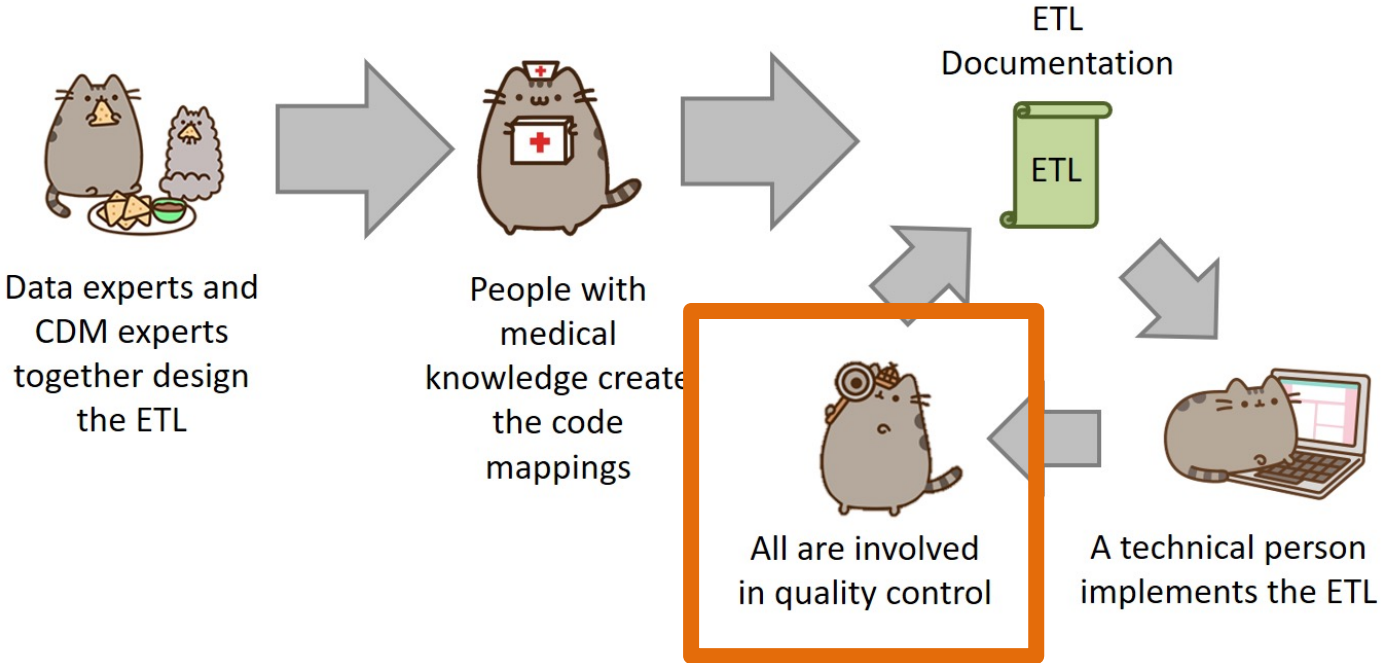
ETL Implementation

General Flow of Implementation





OHDSI
OBSERVATIONAL HEALTH DATA SCIENCES AND INFORMATICS





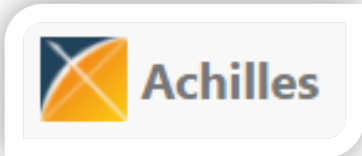
Quality



What tools are available to check that the CDM logic was implemented correctly?



Rabbit-in-a-Hat Test Case Framework



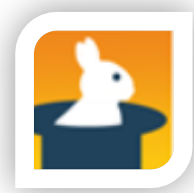
Achilles



DataQualityDashboard (DQD)



Unit Test Cases



- Testing your CDM builder is important:
 - ETL is often complex, increasing the danger of making mistakes that go unnoticed
 - CDM can update
 - Source data structure/contents can change over time
- Rabbit-In-a-Hat can construct unit tests, or small pieces of code that can automatically check single aspects of the ETL design



Achilles



Achilles is a data characterization and quality tool available for download here:

<https://github.com/OHDSI/Achilles>

For an example of how it was run for some sample data, that R script is located here:

<https://github.com/OHDSI/Tutorial-ETL/blob/master/materials/Achilles/achillesRun.R>



DataQualityDashboard (DQD)



- Runs a prespecified set of data quality checks and thresholds on the CDM

DATA QUALITY ASSESSMENT

SYNTHEA SYNTHETIC HEALTH DATABASE

Results generated at 2019-08-22 14:15:06 in 29 mins

	Verification				Validation				Total			
	Pass	Fail	Total	% Pass	Pass	Fail	Total	% Pass	Pass	Fail	Total	% Pass
Plausibility	159	21	180	88%	283	0	283	100%	442	21	463	95%
Conformance	637	34	671	95%	104	0	104	100%	741	34	775	96%
Completeness	369	17	386	96%	5	10	15	33%	374	27	401	93%
Total	1165	72	1237	94%	392	10	402	98%	1557	82	1639	95%

OVERVIEW

METADATA

RESULTS

ABOUT



10 Minute Energy Break

Upon your return, please open the ScanReport sent earlier this week. We will work through it in the next session



White Rabbit



White Rabbit - Location



White Rabbit

Help

Locations Scan Fake data generation

Working folder
C:\ohdsi\WhiteRabbit\WhiteRabbit_v0.8.1\bin Pick folder

Source data location

Data type: Delimited text files

Server location: 127.0.0.1

User name

Password

Database name

Delimiter: ,

Test connection

Console



White Rabbit - Scan



White Rabbit

Help

Locations Scan Fake data generation

Tables to scan

Add all in DB

Add

Remove

Scan field values Min cell count Max distinct values Rows per table

Scan tables

Console



White Rabbit - Scan



White Rabbit

Help

Locations Scan Fake data generation

Tables to scan

Add all in DB

Add

Remove

Scan field values

Min cell count 5

Max distinct values 1,000

Rows per table 100,000

Scan tables

Console



Let's open the WR scan



Other ETL tools

- Excel Macros to Format White Rabbit Scan Report
 - located in OHDSI/sandbox
- Jackalope
- Perseus
- Epic User Web
 - Search “OMOP” as a keyword in the Epic forums

*This is not an exhaustive list. By Googling “OMOP ETL tools” you will find many papers and GitHub repositories devoted to helping you convert your health data to the CDM



OHDSI Community Resources

- GitHub
- forums.ohdsi.org
 - CDM builders
 - Implementers
 - Vocabulary
 - General - introduce yourself on the “Welcome to OHDSI” thread