

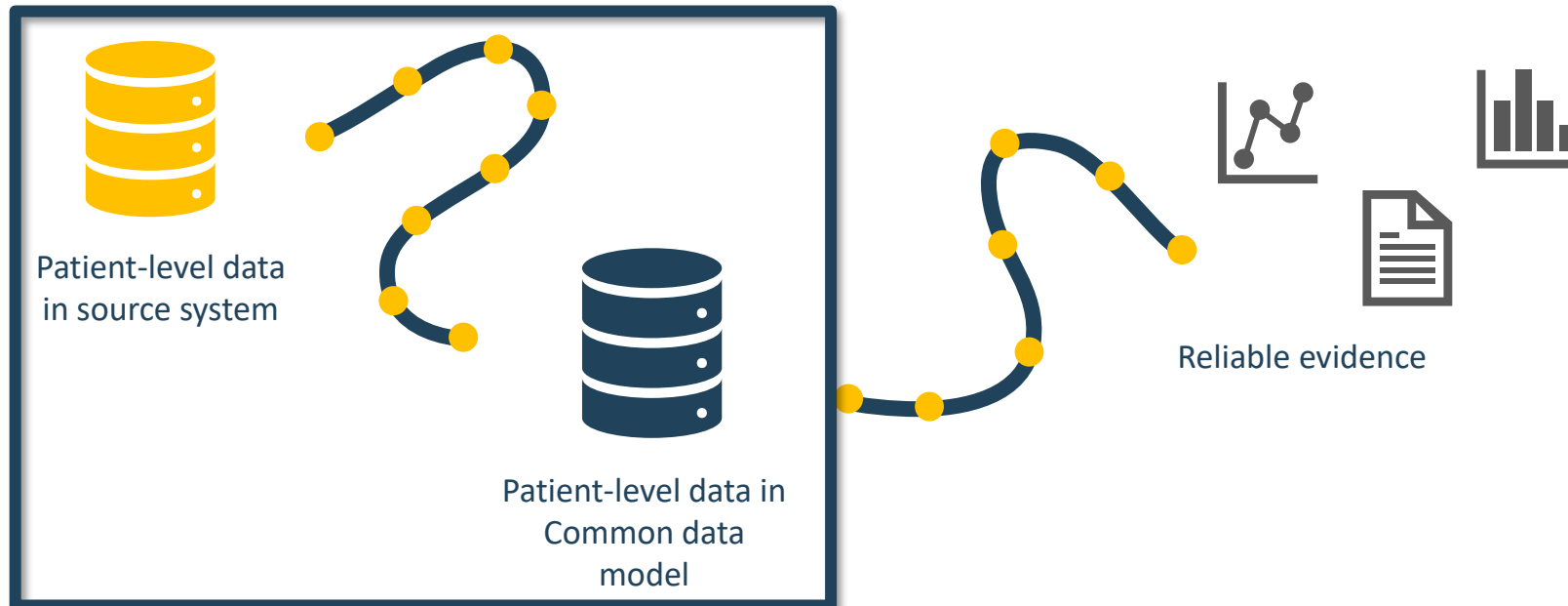


# OMOP Conversion Process



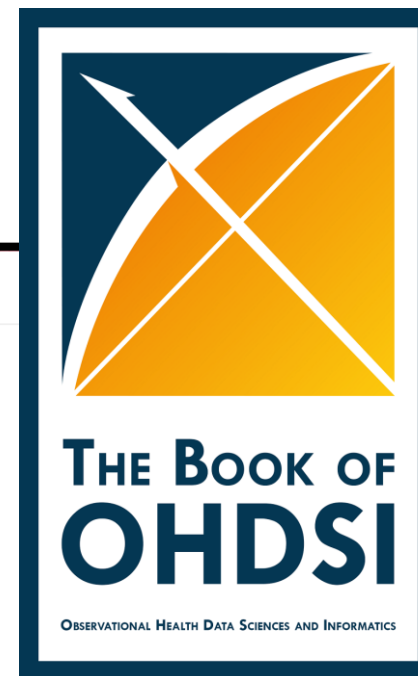
# ETL

- 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 three 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.

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

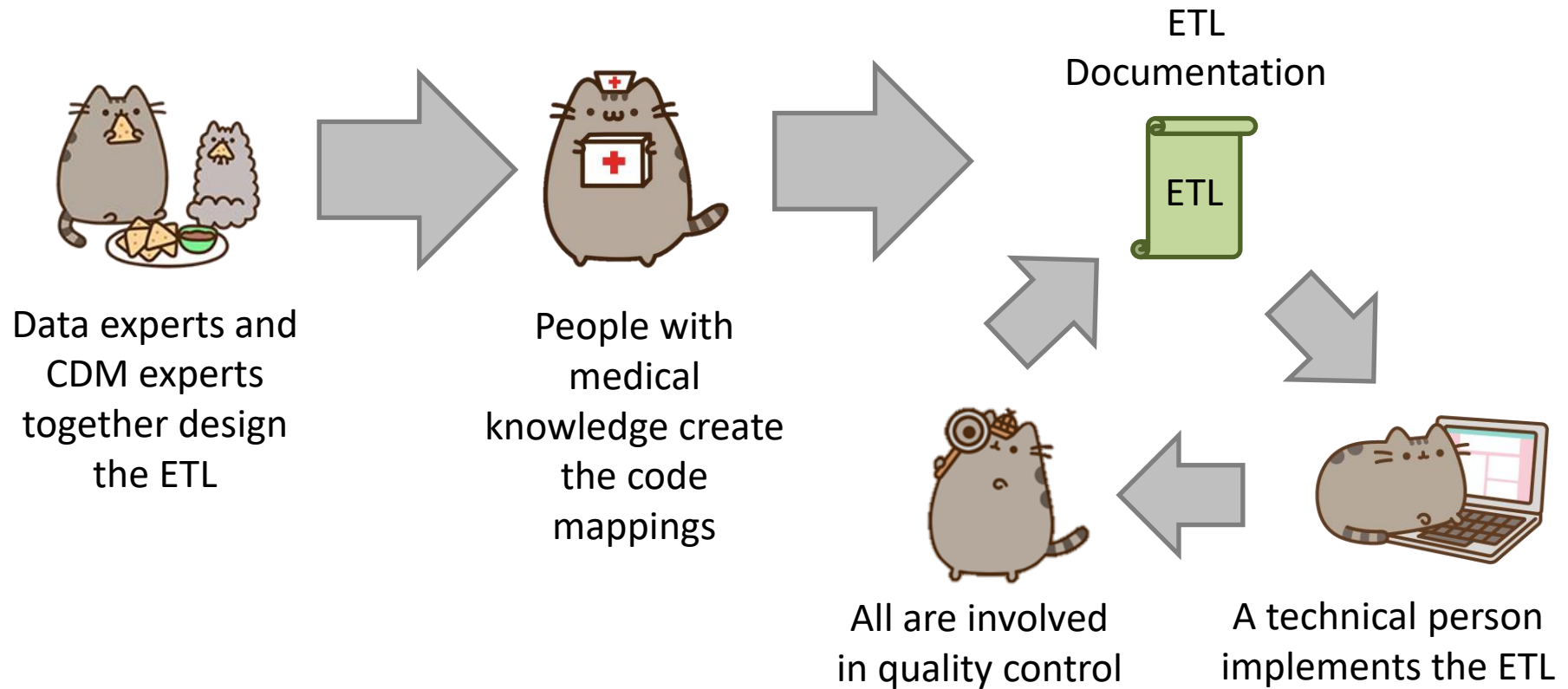
In order to get from the native/raw data to the OMOP Common Data Model (CDM) we have to create an extract, transform, and load (ETL) process. This process should restructure the data to the CDM, and add mappings to the Standardized Vocabularies, and is typically implemented as a set of automated scripts, 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

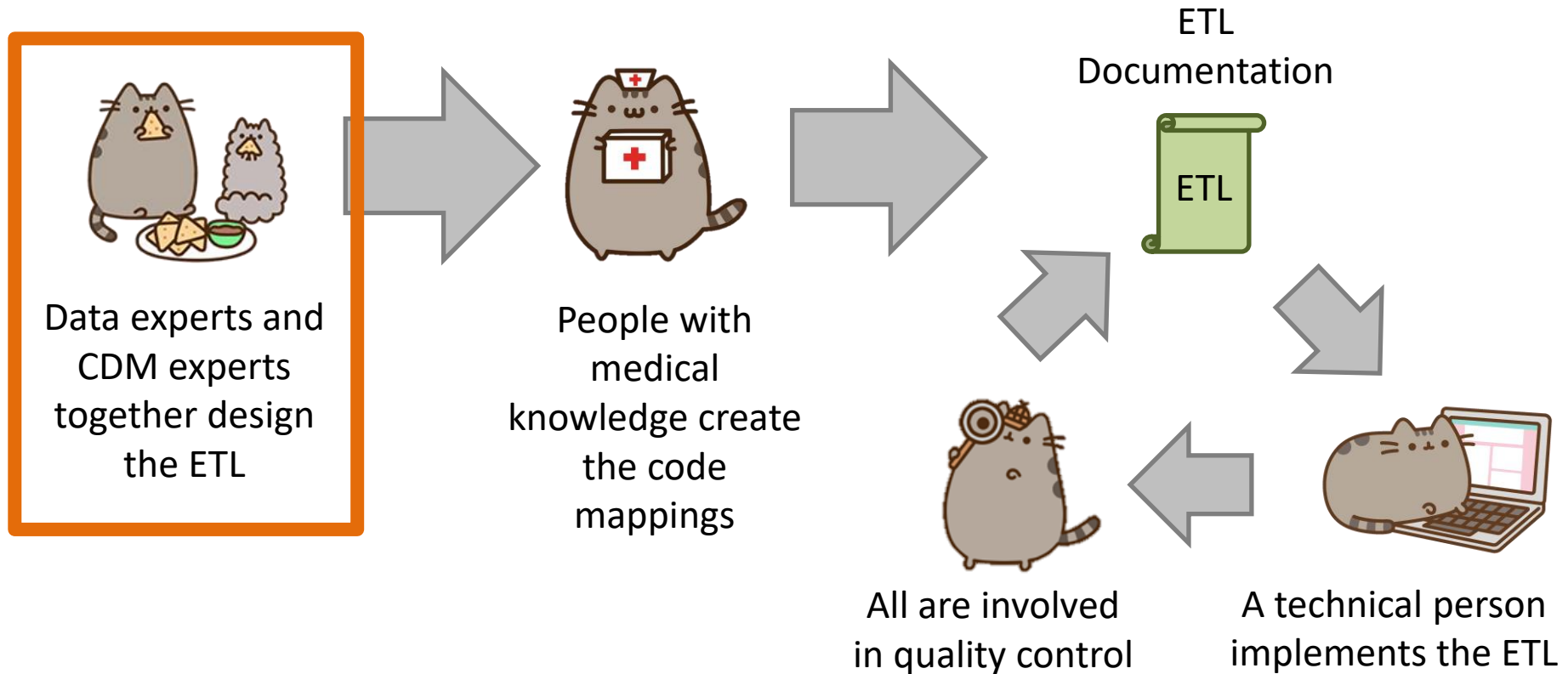


**OHDSI Tools**

- White Rabbit
- Rabbit In a Hat
- Usagi
- White Rabbit
- ACHILLES
- DQD
- Rabbit In a Hat



# Designing the ETL

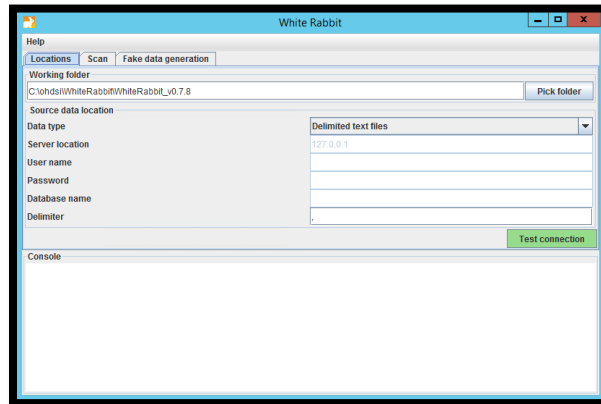




# White Rabbit



- 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	

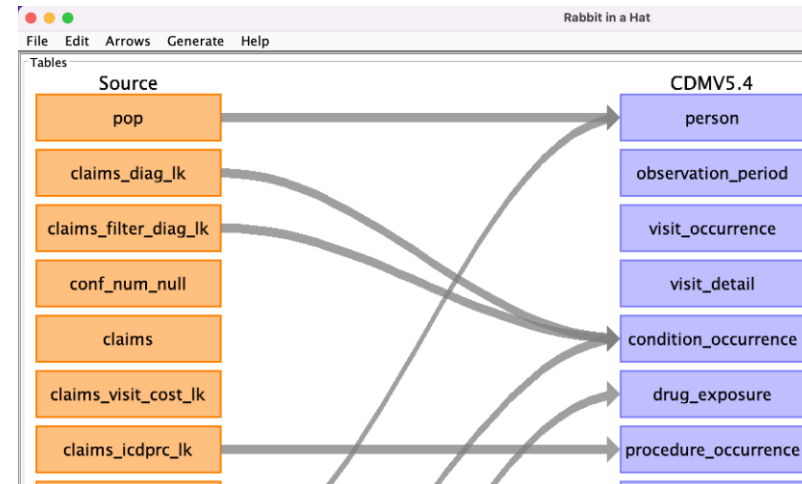
Navigation: pop | claims\_diag\_lk | claims...



# 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

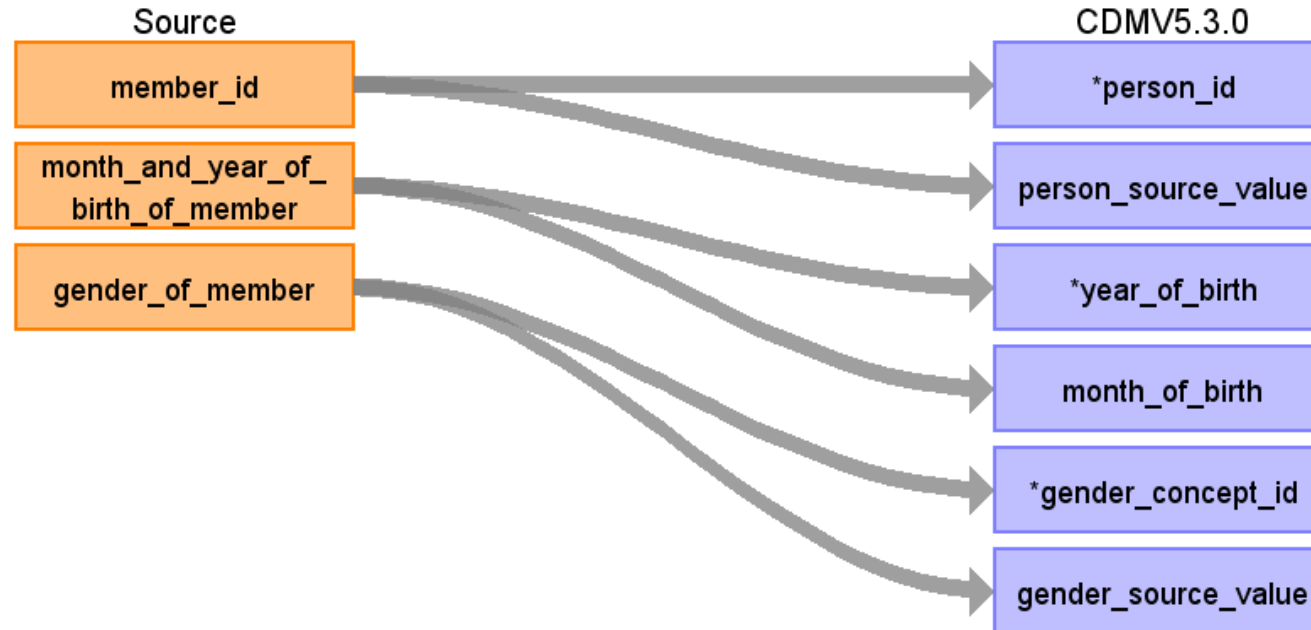






# RIAH – Column Mapping Example

## Reading from Enrollment



Destination Field	Source field	Logic
YEAR_OF_BIRTH	month_and_year_of_birth_of_member	Take first 4 digits
MONTH_OF_BIRTH	month_and_year_of_birth_of_member	Take last 2 digits (01 is January)



# RiaH - Output



## Word document

The screenshot shows a Microsoft Word document with a table and a diagram. The table lists various fields and their values. Below the table, there is a diagram showing the mapping of source fields to destination fields. The diagram shows 'subject\_id' mapping to '\*person\_id', 'date\_diag\_875\_i1' mapping to '\*observation\_concept\_id', and 'history\_solitary\_plasmocyt\_i1' mapping to '\*observation\_date'. Below the diagram is another table with columns for Destination Field, Source Field, Logic, and Comment.

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

## Html

The screenshot shows a web browser displaying the HTML output of the RiaH tool. The page title is 'Person' and it is reading from a Synthea table named 'patients.csv'. The page contains a table with columns for Destination Field, Source field, Logic, and Comment field.

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	

## Markdown documents

The screenshot shows a Markdown document containing the ETL logic for the Person table. The document is written in a code-like format and includes comments and logic for mapping source fields to destination fields.

```
layout: default
title: Person
nav_order: 1
parents: CDM_Synthea_v1
description: "Person mapping from patients.csv"

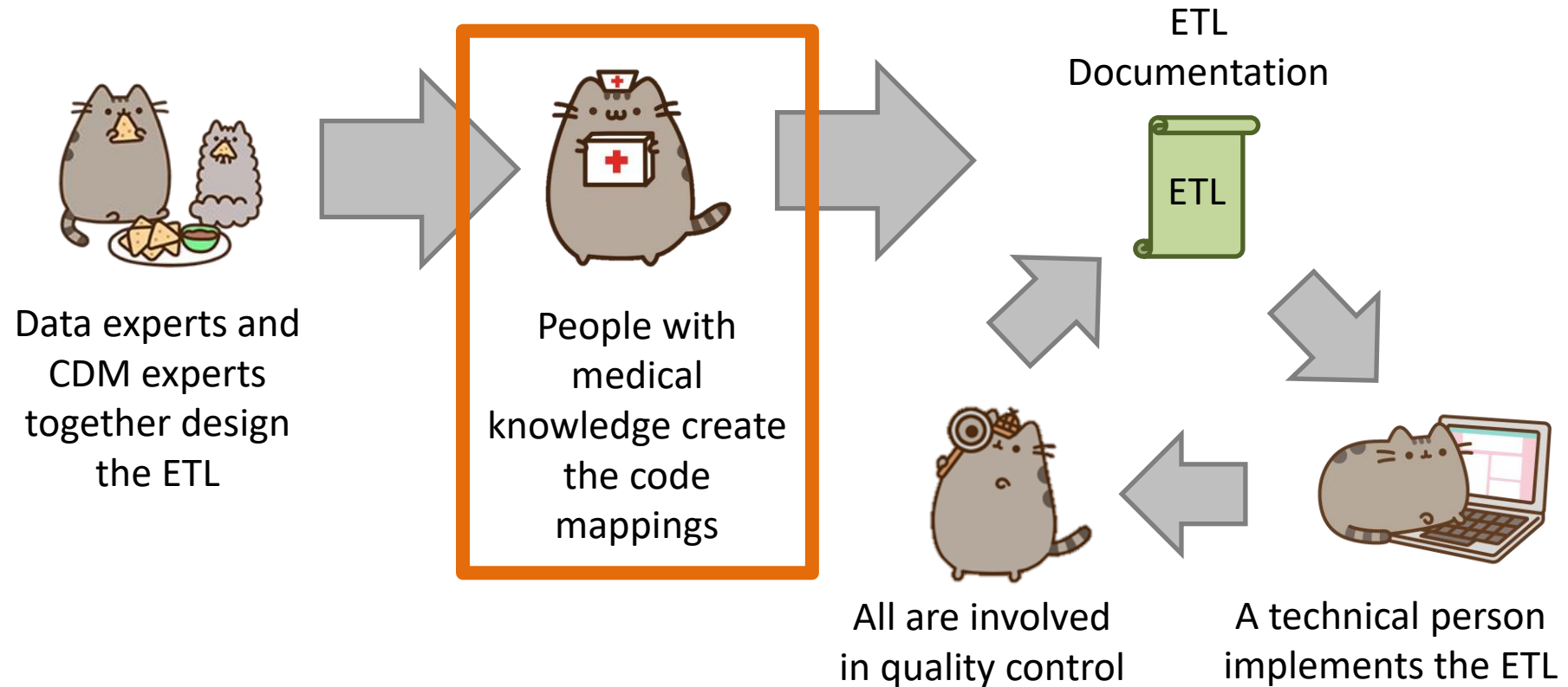
# Person
## Reading from Synthea table patients.csv



| 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 = 'BLACK' then set as 8516, when race = 'ASIAN' then set as 8515, otherwise set as 8517 | |
| ethnicity_concept_id | race | ethnicity | When race = 'HISPANIC', or when ethnicity in ('CENTRAL_AMERICAN', 'DOMINICAN', 'MEXICAN', 'PUERTO_RICAN', 'SOUTH_AMERICAN') then set as 38803563, otherwise set as 0 | |
| location_id | | | |
| provider_id | | | |
| care_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 | | | |
```



# Vocabulary Mapping





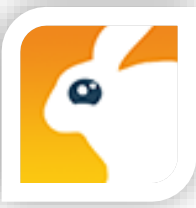
# Using OMOP Vocabularies

Destination Field	Source field	Logic	Comment field
person_id			
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.

Destination Field	Source field	Logic	Comment field
condition_concept_id	code	Use code to lookup target_concept_id in SOURCE_TO_STANDARD_VOCAB_MAP: select v.target_concept_id from conditions c join source_to_standard_vocab_map v on v.source_code = c.code and v.target_domain_id = 'Condition' and v.target_standard_concept = 'S' and v.source_vocabulary_id in ('ICD10')	



# 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 with a menu bar (File, Edit, View, Help) and a main table of source terms and target concepts. Below the table are sections for 'Source code', 'Target concepts', 'Search', and 'Results'.

Status	Source code	Source term	Frequency	ICPC_DES...	Match score	Concept ID	Concept na...	Domain	Concept cl...	Vocabulary	Concept co...	Standard c...	Parents	Children	Comment
Unchecked	A97	No illness	500000	Geen ziekte	0.82	4192174	Illness	Condition	Clinical Fin...	SNOMED	39104002	S	1	3	
Unchecked	S74	Dermatomy...	100000	Dermatomy...	0.81	135473	Dermatoph...	Condition	Clinical Fin...	SNOMED	47382004	S	4	25	
Unchecked	L99	Other disea...	100000	Andere ziek...	0.77	4244662	Disorder of...	Condition	Clinical Fin...	SNOMED	928000	S	3	84	
Unchecked	R74.02	Acute phary...	800000	Acute phary...	1.00	25297	Acute phary...	Condition	Clinical Fin...	SNOMED	363746003	S	6	10	
Unchecked	U71	Cystitis / uri...	500000	Cystitis/urin...	0.71	81902	Urinary trac...	Condition	Clinical Fin...	SNOMED	68566005	S	5	17	
Unchecked	R78.00	Acute bronc...	300000	Acute bronc...	0.84	260125	Acute bronc...	Condition	Clinical Fin...	SNOMED	5505005	S	5	4	
Unchecked	W78.00	Pregnancy ...	100000	Zwangersc...	0.84	4299535	Pregnant	Condition	Clinical Fin...	SNOMED	77386006	S	2	17	
Unchecked	T83.0	overweight	100000	overgewicht	1.00	437525	Overweight	Observation	Clinical Fin...	SNOMED	238131007	S	2	5	
Unchecked	R74	Acute uppe...	800000	Acute infect...	1.00	257011	Acute uppe...	Condition	Clinical Fin...	SNOMED	54398005	S	6	22	
Unchecked	R65.00	episode on...	1	episode op...	0.35	444406	Acute sube...	Condition	Clinical Fin...	SNOMED	70422006	S	4	0	
Unchecked	R44	Immunizati...	1000000	Immunisati...	0.70	4144375	Active imm...	Procedure	Clinical Fin...	SNOMED	33879002	S	2	19	
Unchecked	R05	Cough	880000	Hoesten	1.00	254761	Cough	Condition	Clinical Fin...	SNOMED	49727002	S	2	38	

**Source code**

Source code	Source term	Frequency	ICPC_DESCRIPTION_DUTCH
A97	No illness	500000	Geen ziekte

**Target concepts**

Concept ID	Concept name	Domain	Concept class	Vocabulary	Concept code	Standard concept	Parents	Children
4192174	Illness	Condition	Clinical Finding	SNOMED	39104002	S	1	3

**Search**

Query:

Use source term as query

Query:

**Filters**

Filter by user selected concepts

Filter standard concepts

Include source terms

Filter by concept class:

Filter by vocabulary:

Filter by domain:

**Results**

Score	Term	Concept ID	Concept name	Domain	Concept class	Vocabulary	Concept code	Standard concept	Parents	Children
0.82	Illness	4192174	Illness	Condition	Clinical Finding	SNOMED	39104002	S	1	3
0.80	Mental illness	4214703	Mental illness	Observation	Qualifier Value	SNOMED	394816006	S	1	0
0.80	Mental illness	432586	Mental disorder	Condition	Clinical Finding	SNOMED	74732009	S	2	41
0.78	Viral illness	440029	Viral disease	Condition	Clinical Finding	SNOMED	34014006	S	3	31
0.77	Mass illness	45883959	Mass illness	Meas Value	Answer	LOINC	LA18096-0	S	0	0
0.75	Stillness	4092256	Stillness	Condition	Clinical Finding	SNOMED	247902008	S	3	1

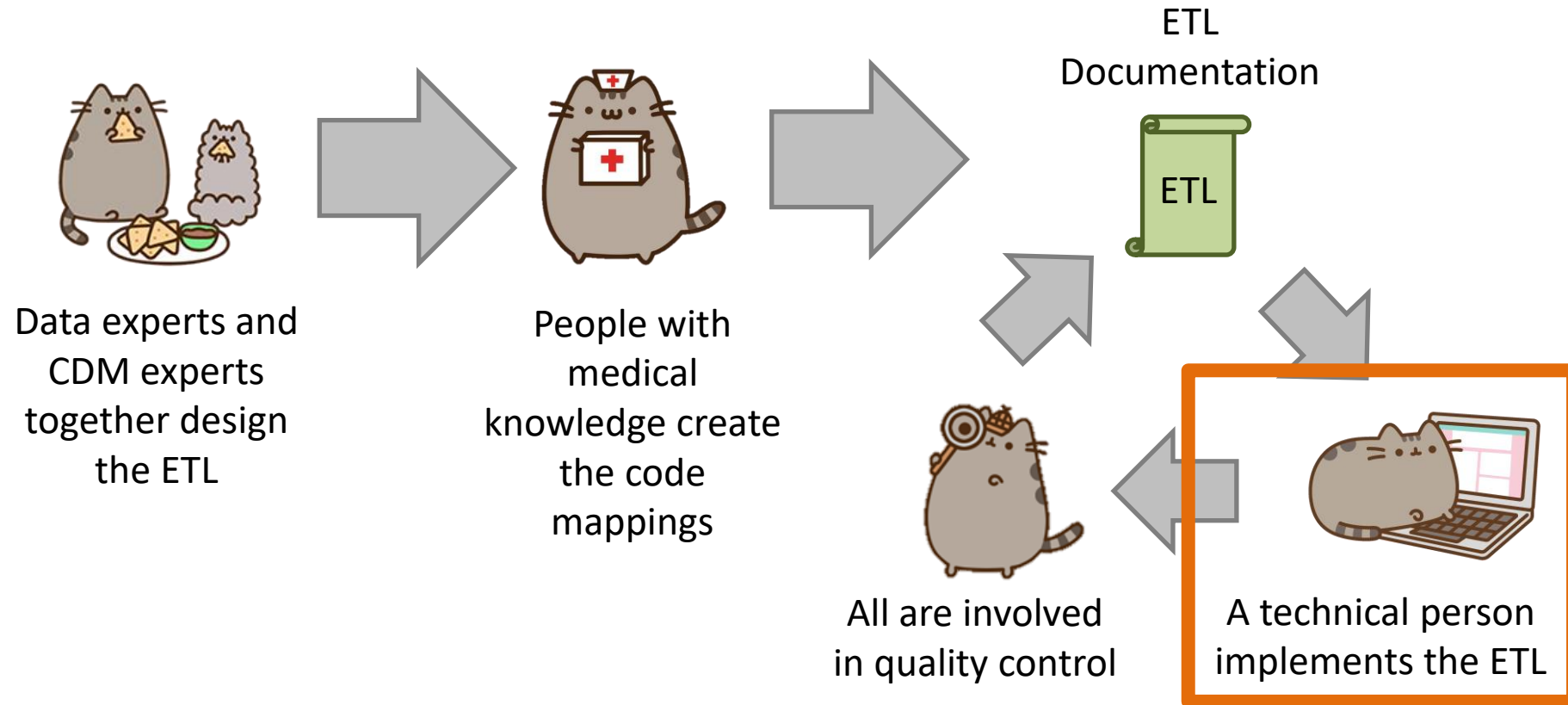
Comment:

Approved / total: 0 / 12 0.0% of total frequency

Vocabulary version: v5.0 19-NOV-18



# Implementing the ETL





# ETL Implementation



There are multiple tools available to implement your ETL

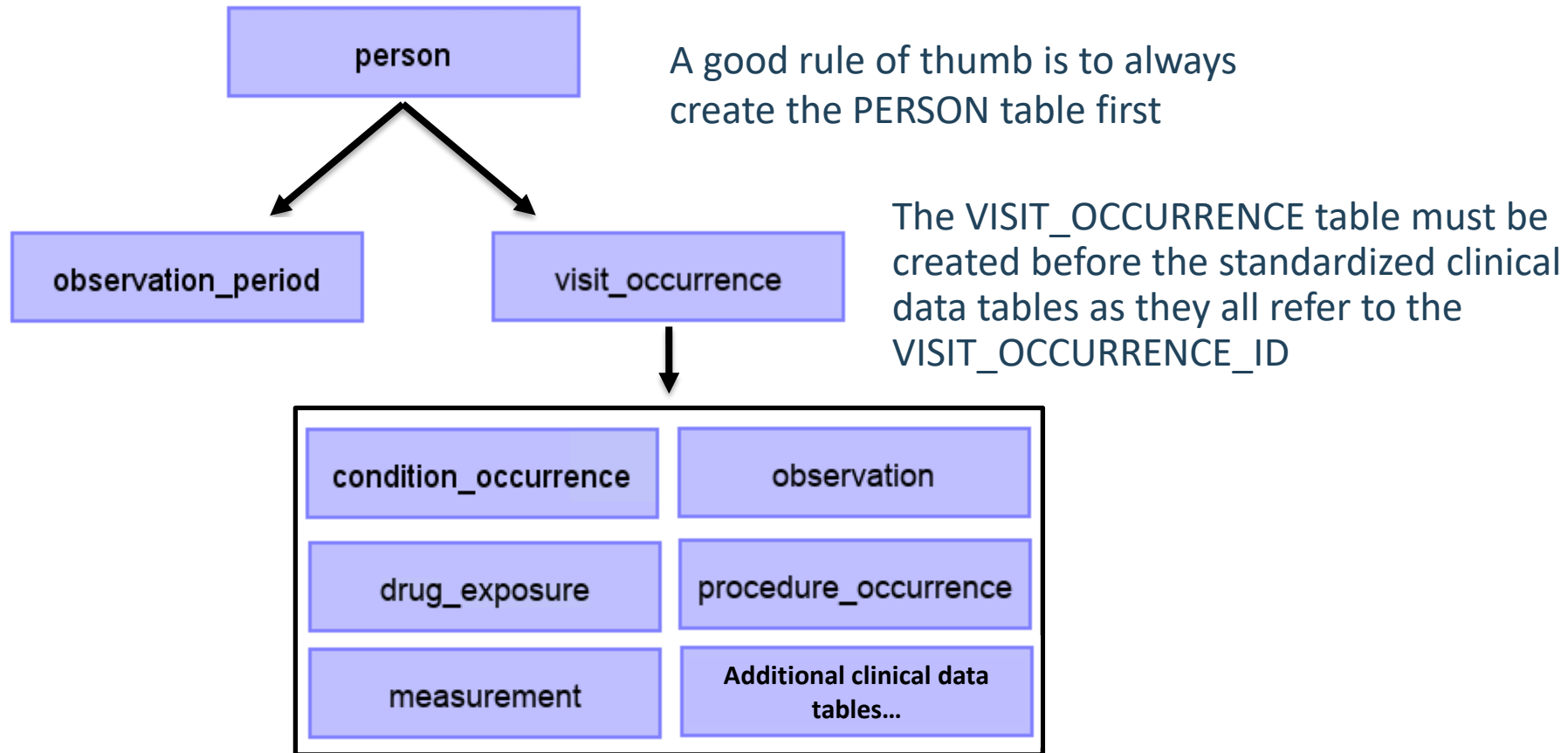


Your choice will largely depend on the size and complexity of the ETL design. And the tools available to you.



# ETL Implementation

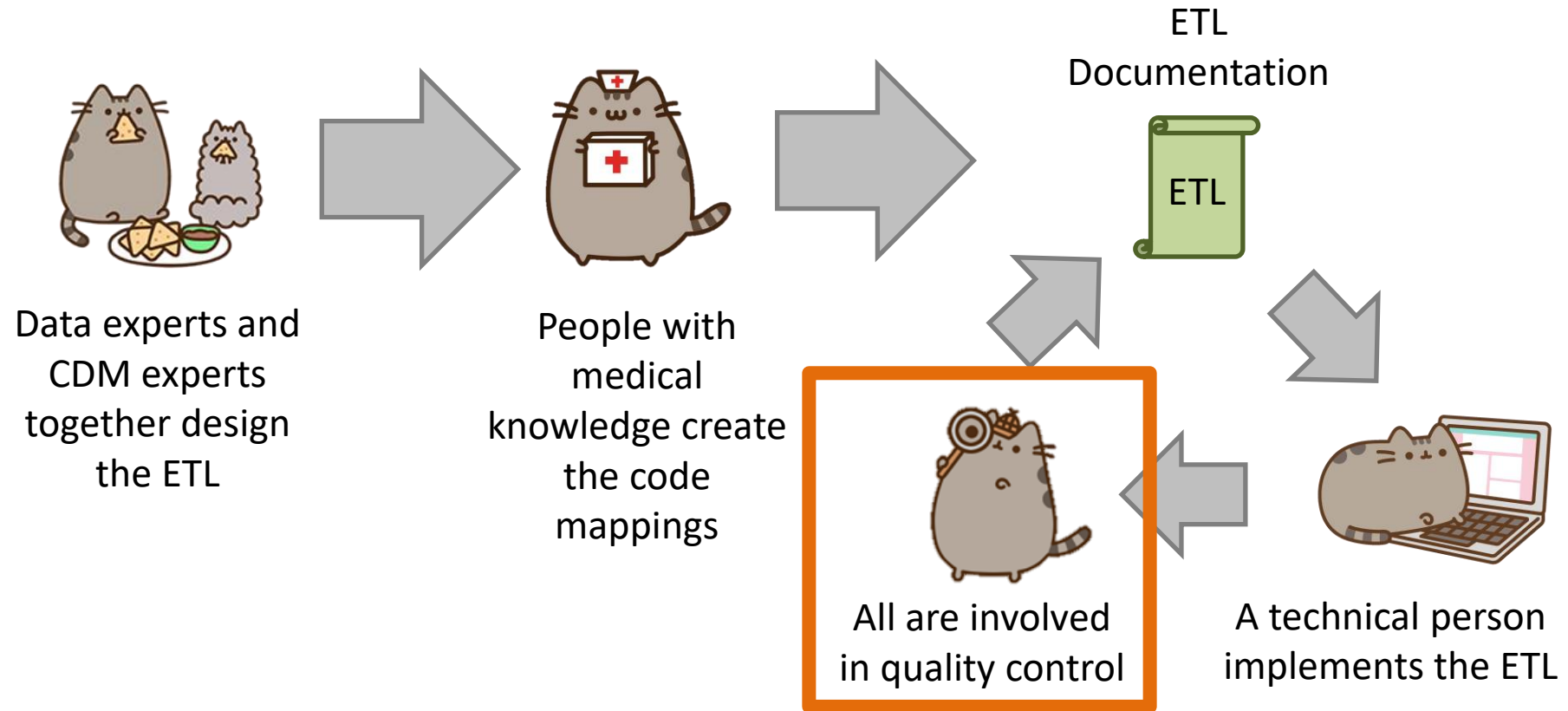
## General Flow of Implementation







# Quality Control





# Quality



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



Rabbit-in-a-Hat Test Case Framework



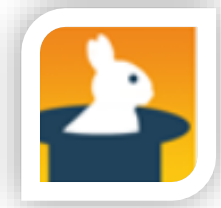
Achilles



DataQualityDashboard (DQD)



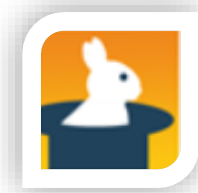
# Unit Test Cases



- 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



# Unit Test Cases

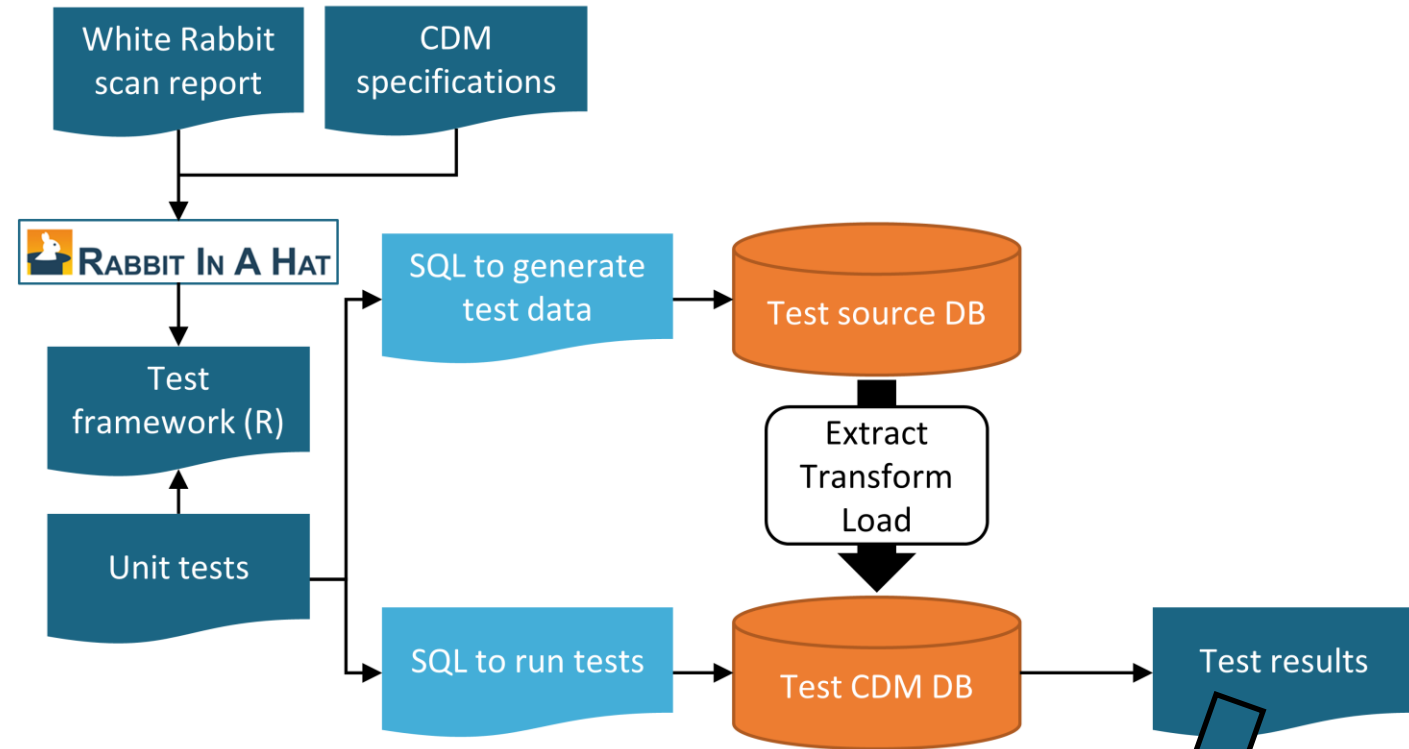
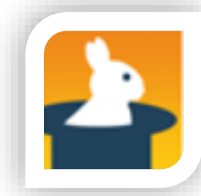


The test framework creates a series of R functions that enables you to specify your ‘fake’ people and records in the same structure as your source data using the scan report as a guide.

```
source("Framework.R")
declareTest(101, "Person gender mappings")
add_enrollment(member_id = "M000000102", gender_of_member = "male")
add_enrollment(member_id = "M000000103", gender_of_member = "female")
expect_person(PERSON_ID = 102, GENDER_CONCEPT_ID = 8507)
expect_person(PERSON_ID = 103, GENDER_CONCEPT_ID = 8532)
```



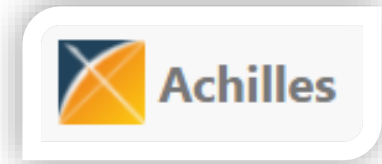
# Unit Test Cases



ID	Description	Status
101	Person gender mappings	PASS
101	Person gender mappings	PASS



# Achilles

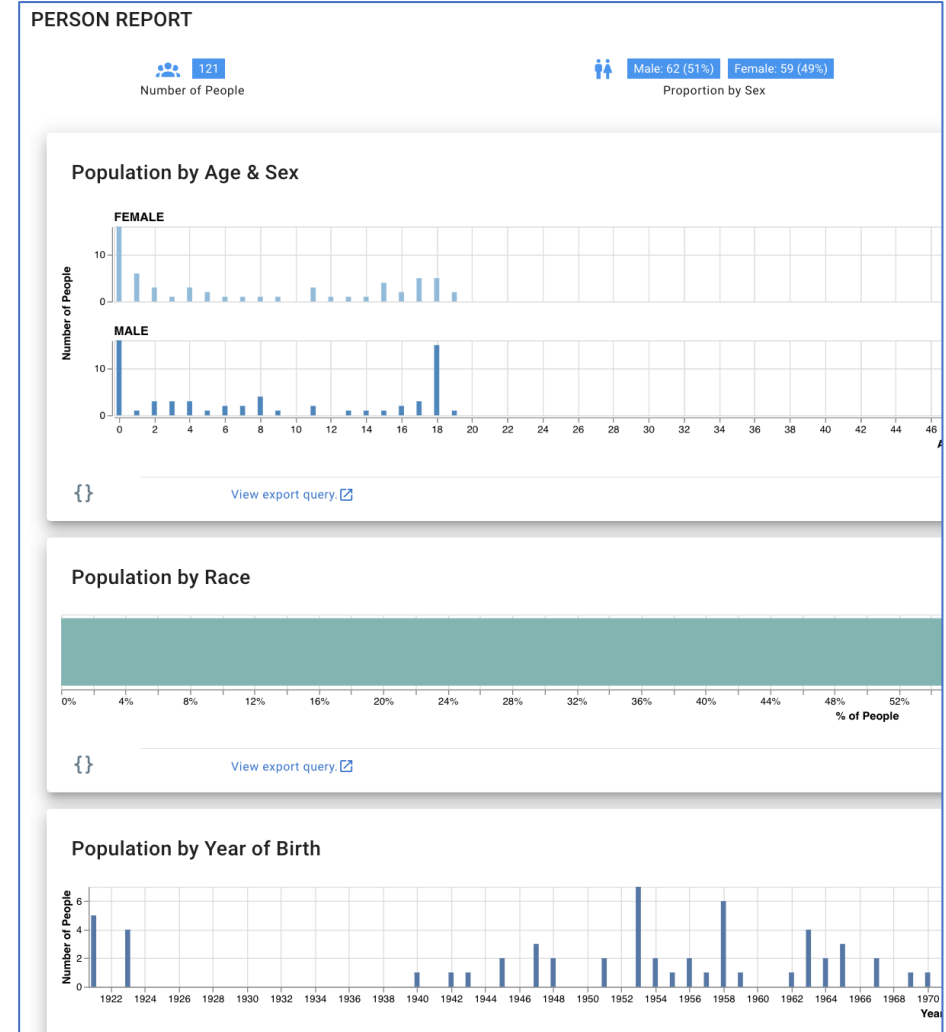


Achilles is a data characterization and quality tool available for download here:

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

Provides descriptive statistics on an OMOP CDM

Results can be visualized in ARES or ATLAS

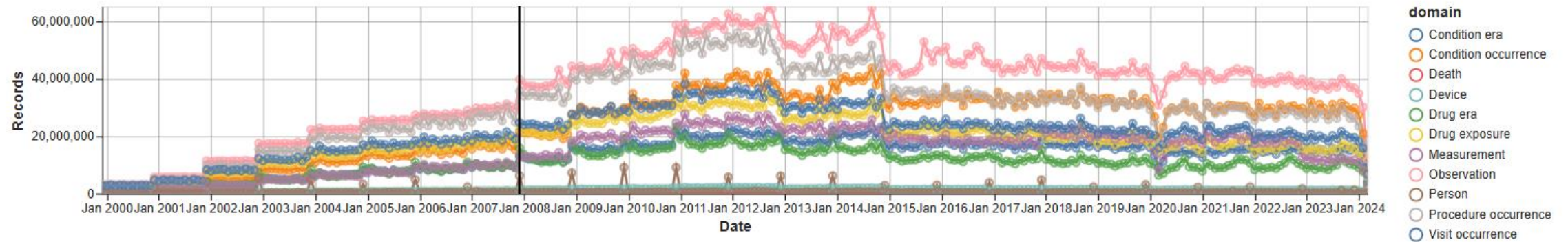




# ARES: Data Density Plot



## DOMAIN DENSITY

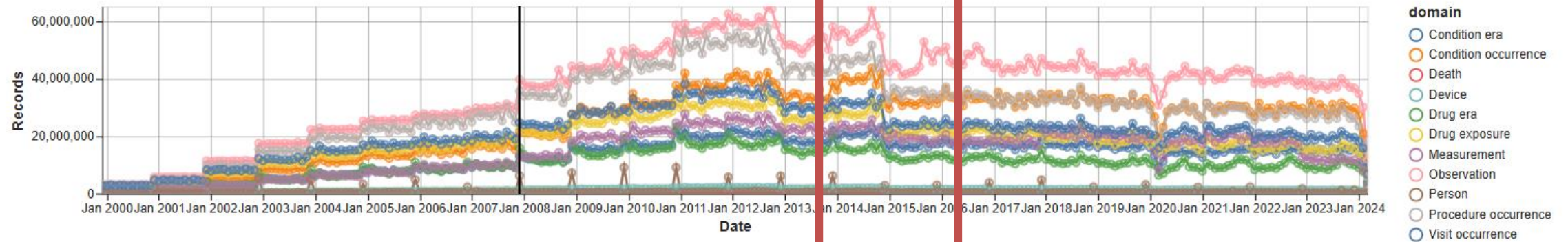




# ARES: Data Density Plot



## DOMAIN DENSITY



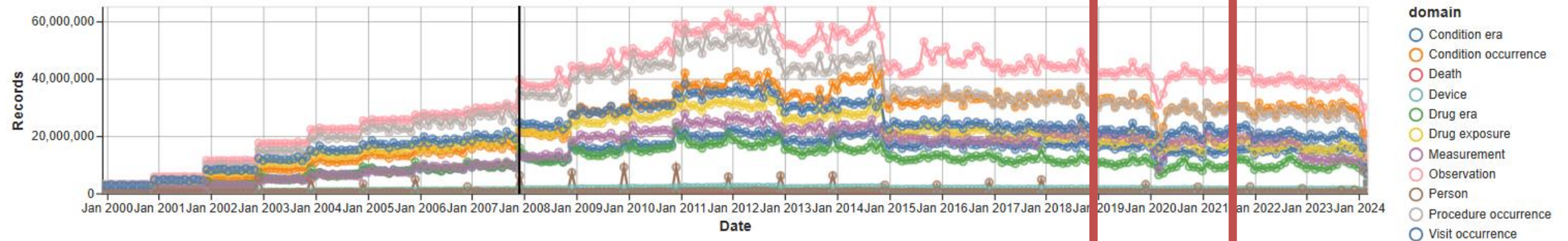




# ARES: Data Density Plot



## DOMAIN DENSITY

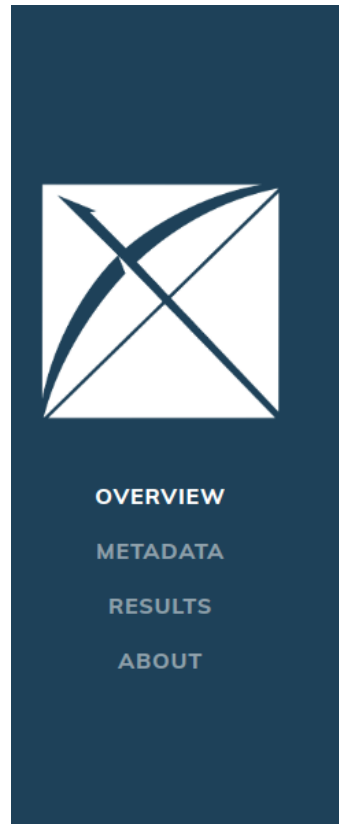




# DataQualityDashboard (DQD)



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



## DATA QUALITY ASSESSMENT

### SYNTHETA 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	<b>21</b>	180	88%	283	0	283	100%	442	<b>21</b>	463	95%
Conformance	637	<b>34</b>	671	95%	104	0	104	100%	741	<b>34</b>	775	96%
Completeness	369	<b>17</b>	386	96%	5	<b>10</b>	15	33%	374	<b>27</b>	401	93%
Total	1165	<b>72</b>	1237	94%	392	<b>10</b>	402	98%	1557	<b>82</b>	1639	<b>95%</b>



# DQD Example Rules

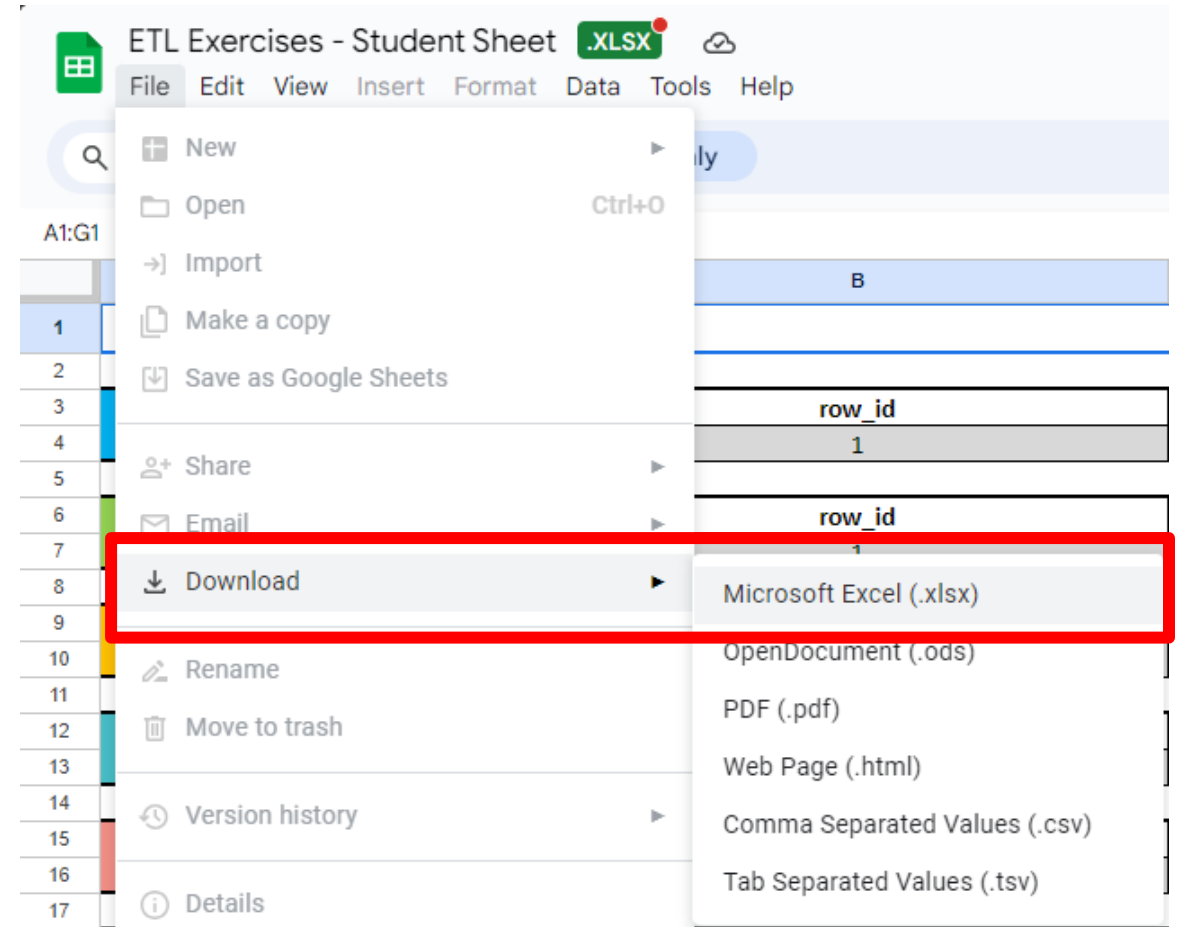
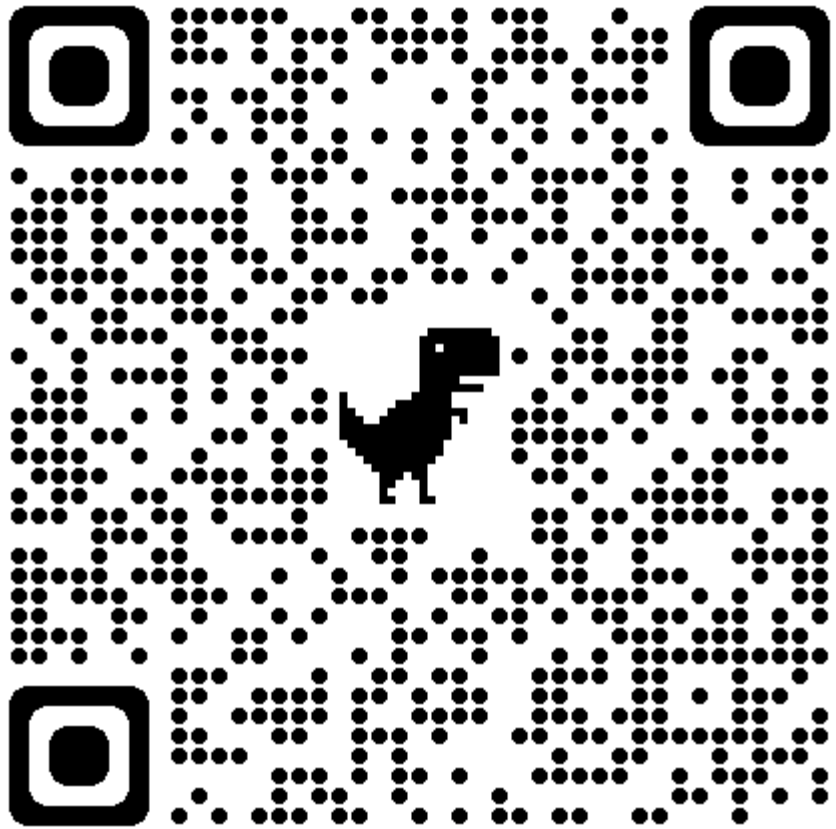


Fraction violated rows	Check description	Threshold	Status
0.34	A yes or no value indicating if the provider_id in the VISIT_OCCURRENCE is the expected data type based on the specification.	0.05	FAIL
0.99	The number and percent of distinct source values in the measurement_source_value field of the MEASUREMENT table mapped to 0.	0.30	FAIL
0.09	The number and percent of records that have a value in the drug_concept_id field in the DRUG_ERA table that do not conform to the ingredient class.	0.10	PASS
0.02	The number and percent of records with a value in the verbatim_end_date field of the DRUG_EXPOSURE that occurs prior to the date in the DRUG_EXPOSURE_START_DATE field of the DRUG_EXPOSURE table.	0.05	PASS
0.00	The number and percent of records that have a duplicate value in the procedure_occurrence_id field of the PROCEDURE_OCCURRENCE.	0.00	PASS



# Exercise Instructions

- Download a copy of the exercises at:





# Exercise Instructions

- Together as a group, we will map the native data provided to the OMOP CDM using the template provided in the *ETL Development\_1000* sheet
- You will then be given time to do the same on your own for the *ETL Development\_1005* and *ETL Development\_1010* sheets



**Thank you!**