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
Chapter 6  Extract Transform Load

Chapter leads: Clair Biacketer & 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.
Data experts and CDM experts together design the ETL.

People with medical knowledge create the code mappings.

All are involved in quality control.

A technical person implements the ETL.

ETL Documentation

ETL
Designing the ETL

Data experts and CDM experts together design the ETL.

People with medical knowledge create the code mappings.

All are involved in quality control.

A technical person implements the ETL.

ETL Documentation
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
### Table/Field Overview

<table>
<thead>
<tr>
<th>Table</th>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Max length</th>
<th>N rows</th>
</tr>
</thead>
<tbody>
<tr>
<td>pop</td>
<td>der_sex</td>
<td>character</td>
<td>1</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>der_yob</td>
<td>double</td>
<td>6</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>pat_id</td>
<td>character</td>
<td>64</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>pat_hash_id</td>
<td>character</td>
<td>16</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>pmtx_flag</td>
<td>numeric</td>
<td>1</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>anon_ims_pat_id</td>
<td>character</td>
<td>11</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>pat_region</td>
<td>character</td>
<td>2</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>pat_state</td>
<td>character</td>
<td>2</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>pat_zip3</td>
<td>character</td>
<td>3</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>grp_indv_cd</td>
<td>character</td>
<td>1</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>mh_cd</td>
<td>character</td>
<td>1</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>enr_rel</td>
<td>character</td>
<td>2</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>temp_col1</td>
<td>character</td>
<td>0</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>temp_col2</td>
<td>character</td>
<td>0</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>pop</td>
<td>load_row_id</td>
<td>bigint</td>
<td>9</td>
<td>16374539</td>
<td></td>
</tr>
<tr>
<td>claims_diag lk</td>
<td>person_source_val</td>
<td>character</td>
<td>64</td>
<td>2992046684</td>
<td></td>
</tr>
<tr>
<td>claims_diag lk</td>
<td>event_start_date</td>
<td>date</td>
<td>10</td>
<td>2992046684</td>
<td></td>
</tr>
<tr>
<td>claims_diag lk</td>
<td>event_end_date</td>
<td>date</td>
<td>10</td>
<td>2992046684</td>
<td></td>
</tr>
</tbody>
</table>

### Value counts

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>der_sex</td>
<td>50479</td>
<td>991.0</td>
<td>2030</td>
<td>Li</td>
</tr>
<tr>
<td>der_yob</td>
<td>49514</td>
<td>992.0</td>
<td>1970</td>
<td></td>
</tr>
<tr>
<td>pat_id</td>
<td>7</td>
<td>1000.0</td>
<td>1947</td>
<td></td>
</tr>
<tr>
<td>pat_hash_id</td>
<td>1094.0</td>
<td>1908</td>
<td>1873</td>
<td></td>
</tr>
<tr>
<td>pat_region</td>
<td>1094.0</td>
<td>1908</td>
<td>1873</td>
<td></td>
</tr>
<tr>
<td>pat_state</td>
<td>1094.0</td>
<td>1908</td>
<td>1873</td>
<td></td>
</tr>
<tr>
<td>pat_zip3</td>
<td>1094.0</td>
<td>1908</td>
<td>1873</td>
<td></td>
</tr>
<tr>
<td>grp_indv_cd</td>
<td>1094.0</td>
<td>1908</td>
<td>1873</td>
<td></td>
</tr>
<tr>
<td>mh_cd</td>
<td>1094.0</td>
<td>1908</td>
<td>1873</td>
<td></td>
</tr>
<tr>
<td>enr_rel</td>
<td>1094.0</td>
<td>1908</td>
<td>1873</td>
<td></td>
</tr>
<tr>
<td>temp_col1</td>
<td>1094.0</td>
<td>1908</td>
<td>1873</td>
<td></td>
</tr>
<tr>
<td>temp_col2</td>
<td>1094.0</td>
<td>1908</td>
<td>1873</td>
<td></td>
</tr>
<tr>
<td>load_row_id</td>
<td>1094.0</td>
<td>1908</td>
<td>1873</td>
<td></td>
</tr>
</tbody>
</table>

- Claims diag lk: person_source_val, event_start_date, event_end_date
• Read and display a White Rabbit scan document

• Provides a graphical interface to allow a user to connect source data to CDM tables
Vocabulary Mapping

Data experts and CDM experts together design the ETL.

People with medical knowledge create the code mappings.

All are involved in quality control.

A technical person implements the ETL.

ETL Documentation
• 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’

Usagi
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
Implementing the ETL

Data experts and CDM experts together design the ETL

People with medical knowledge create the code mappings

All are involved in quality control

A technical person implements the ETL
ETL Implementation

There are multiple tools available to implement your ETL

R

Python

PostgreSQL

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

A good rule of thumb is to always create the PERSON table first.

The VISIT_OCCURRENCE table must be created before the standardized clinical data tables as they all refer to the VISIT_OCCURRENCE_ID.
Data experts and CDM experts together design the ETL

People with medical knowledge create the code mappings

All are involved in quality control

A technical person implements the ETL
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:

DataQualityDashboard (DQD)

• Runs a prespecified set of data quality checks and thresholds on the CDM
Common ETL Issues

- **Non-standard Vocabulary**
  Codes mapped to OMOP vocabulary aren’t mapped to a ‘Standard’

- **Multiple Input on Records**
  Some records will contain multiple coding systems and text. A hierarchy must be selected to avoid duplicate records

- **Multiple records for one concept mapping**
  Picking one of the multiple standard vocabulary mapping to create the OMOP CDM record instead of one record per mapping

- **Wrong type_concept_id**
  Use of the wrong type_concept_id or misunderstanding the definition of this field

- **Abnormal values**
  Unconventional values in data asset (i.e. Negative or 0 as value_as_number)

- **Missing CDM tables**
  OMOP CDM tables missing due to misunderstanding on how to populate the table.

- **Non-Clinical Events**
  Due to text options in EHR Data, many options are not clinical events (e.g. ‘Tuesday’ or ‘XXYZ’). These records will be scrubbed to ensure quality of data converted to OMOP.

- **Incorrect logic - Observation_Period**
  Observation_Period table populated incorrectly. Observation period does not cover the entire period of time where events are recorded for a person

- **Incorrect logic - Visit_Occurrence**
  Visit_Occurrence table populated incorrectly
Exercise Instructions

• Download a copy of the exercises at:
Exercise Instructions

• Using the native data provided, map it to the OMOP CDM using the template provided in the *ETL Development_1000* sheet

• If you have spare time, do the same for the *ETL Development_1005* and *ETL Development_1010* sheets
Thank you!

Mui Van Zandt  mui.vanzandt@iqvia.com
Gyeol Song     gyeol.song@iqvia.com