Challenges and solutions in using OMOP CDM to FAIRify a Dutch ICU quality registry

PRESENTER: **Daniel** Püttmann

INTRO:

- **Who cares?** Anyone wanting to adopt the OMOP CDM who wants to know the issues you might encounter during the Extract, Transform and Load (ETL) process.
- **Why?** The COVID-19 pandemic has shown the need for data that is Findable, Accessible, Interoperable and Reusable (FAIR). A popular solution to make databases FAIR is OHDSI's OMOP CDM. Right now, an ETL to the OMOP Common Database Model (CDM) is less of a cakewalk and more of an arduous journey. We made an inventory of issues that need to be addressed during the ETL, and how to solve them.

METHODS

1. We mapped our data elements to the appropriate OMOP CDM fields, and we standardized them using the LOINC and SNOMED-CT standardized vocabularies.
2. We made SQL scripts to execute the ETL as planned in the previous step.
3. We validated the ETL using data quality tools provided by OHDSI.
4. We published our OMOP CDM database on the EHDEN Portal so researchers can find it.

The steps followed are illustrated in **Figure 2.**

RESULTS

Results in **Figure 1.**

We found 4 **challenges** in the **transition** from a national database model to the **OMOP Common Database Model:**

	Challenge	Solution
I. Mapping problems	50 unmapped out of 158 data elements due to: 1. Specific meaning 2. It being a specialized code set	Hybrid solution from a conference poster: 1. Inserting source codes or headers in relevant tables, and then using a lookup table to insert custom codes in those tables. 2. Adding custom codes to the OMOP CDM concept tables and create relationships such as 'maps to' in the concept relationship table.
II. Differences in unit of observation	The NICE database is centered around a patient's admission to the ICU while OMOP CDM is patient oriented. This caused problems with defining primary keys.	We eventually found a way to extract singular patients out of our database using compound keys of various data elements.
III. Differences in database structure	NICE database was in a wide format, while the OMOP CDM uses a long format.	The NICE columns had to be extracted to a temporary table so that they could be pivoted using the 'UNPIVOT' SQL operator, before being inserted into the OMOP CDM tables. See Figure 3 for a visual explanation.
IV. Technical Restrictions	1. Each installation of OHDSI's applications and their dependencies needed to be approved by our IT department. 2. During the ATLAS new tables had to be generated. We could not find a script for this in the Book of OHDSI, EHDEN academy, OHDSI's github, or OHDSI's wiki.	1. Approval by our IT department delayed our process, especially since the tools required specific versions of dependencies, and needed to be reinstalled multiple times. 2. We found a script generating 104 new tables required for ATLAS on the OHDSI forums. The tables generated by the script had redundant primary keys causing issues, so we had to delete them, per instruction by another OHDSI forum post.

Figure 1. The 4 challenges to be addressed during the ETL to the OMOP CDM

Scan QR to **access** the NICE's **EHDEN Portal** page

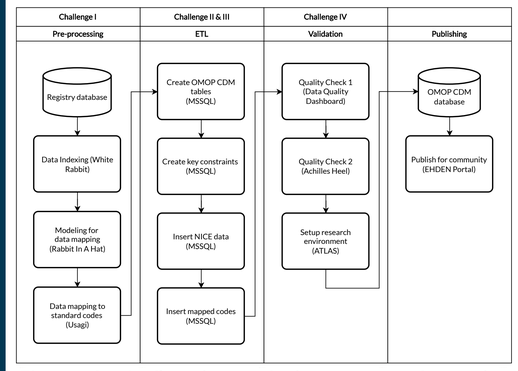

Figure 2. A flowchart of the process from the registry database to the OMOP CDM database. The tools used are mentioned in parentheses. All tools except MSSQL are developed by OHDSI.

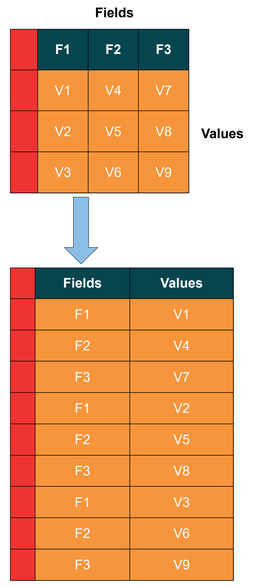

Figure 3. A visual example of an unpivot operation

Daniel Puttmann,
Nicolette de Keizer,
Ronald Cornet,
Eric van der Zwan,
Ferishta Bakhshi-Raiez

Amsterdam UMC
University Medical Centers

OHDSI