U.S. Department of Veterans Affairs

VA Informatics and Computing Infrastructure

Use and Aggregation of OMOP Standard Vocabularies for Administrative Codes

(and a brief mention of Medications): Example building a basic computable phenotype

Michael E. Matheny, MD, MS, MPH

@MichaelEMatheny

Many thanks to many hands!

VINCI Governance Board

- David Atkins, Jack Bates, James Breeling, Stephen Finn, Denise Hynes, Jonathan Nebeker, John Quinn

VINCI OMOP Team

- Scott Duvall, Michael Matheny, Steve Deppen, Kristine Lynch, Jesse Brannen, Aize Cao, Jiwon Chang, Jason Denton, Elizabeth Hanchrow, Abigail Hillard, Kevin Malohi, Daniel Park, Kushan Hewa, Ben Viernes

VINCI IT Team

- Augie Turano, Hamid Saoudian, VINCI IT Administrators

Contributing Investigators

- Makoto Jones, Valmeek Kudesia, Steve Luther, Brian Sauer, Andy Zimolzak

MVP Phenotyping Core

- Kelly Cho, David Gagnon, Jacqueline Honerlaw

Champions and Beta Testers

- Drs. Whooley, Frey, Aragam, Callaway-Lane, Cohen, Grogan, Hwang, Jeffrey, Lenert, Luther, Morano, Mudumbai, Samore, Van Houtven, Voils, Zeng

Poll #1: Your role as a data user

What is your role in research and/or quality improvement?

- Research investigator
- Methodologist
- Data manager, analyst, or programmer
- Project coordinator
- Other – please describe via the Q&A function

Poll #2: Your experience with CDW data

Rate your level of experience with CDW data on a scale of 1 to 5…

- 1. Not worked with it at all
- 2. Have minimal experience with it
- 3. Have work closely with it for <6 months
- 4. Have worked closely with it for 6 months to 2 years
- 5. Very experienced with CDW

Poll #3: Your experience with OMOP data

How familiar are you with OMOP?

- 1. I currently use it
- 2. I would like to use it
- 3. I have heard of it and would like to learn more
- 4. I have not heard of it
- 5. I have my doubts

Objectives

- Setup the Phenotyping Use Case Example
- Brief OMOP Overview & Access Request
- OMOP Architecture with an Administrative Code Focus
- Building an Administrative Code “Filter List” in OMOP
- Applying Code Filter List to Build a Basic Computable Phenotype
- Highlighting medication mapping and support functions

Use Case: Problem Statement

- So you want to find patients in the VA with heart failure…
- “Clinical Phenotype” is the new Cohort Definition
- How to think about building a phenotype that is re-usable across multiple use cases
  - Most use cases apply cohort or phenotype definitions against specific index dates
- For This Example, We Are Limiting to Administrative Code Definitions

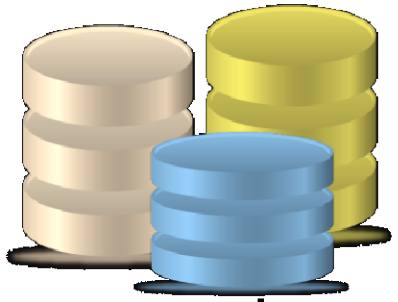
Use Case: Calculate a Cohort/Phenotype

- The Go definition for ***Heart Failure*** used the first single inpatient ICD9 code in the primary position (1st) or the second inpatient non-primary or outpatient primary or secondary ICD9 code to be defined as having Heart Failure.

- The adapted ICD-9 code list used the Go Circulation list with recommended additions from the Mini-Sentinel HF HOI systematic review references.
  - 398.91, 428.*, 402.01, 402.11, 402.91
  - 404.01, 404.03, 404.11, 404.13, 404.91, 404.93

- QUESTIONS
  - How and why is OMOP useful and efficient to use for this type of task?
  - How do you translate a list of ICD-9 codes to OMOP CONCEPT_ID's?
  - How do you use the resulting list o OMOP CONCEPT_ID's to find Heart Failure and establish the time windows of the condition?

What is VA OMOP?

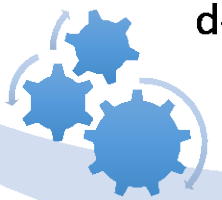


source data

quality assurance

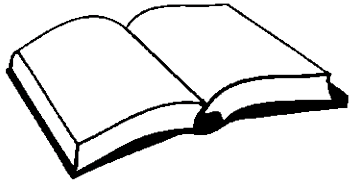
community use and contribution

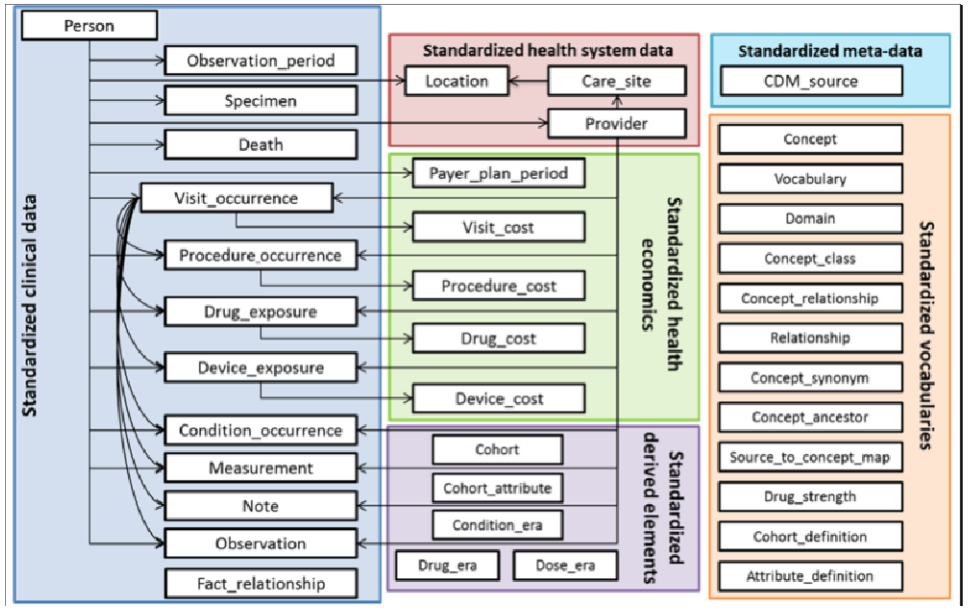
extracted, transformed, derived data



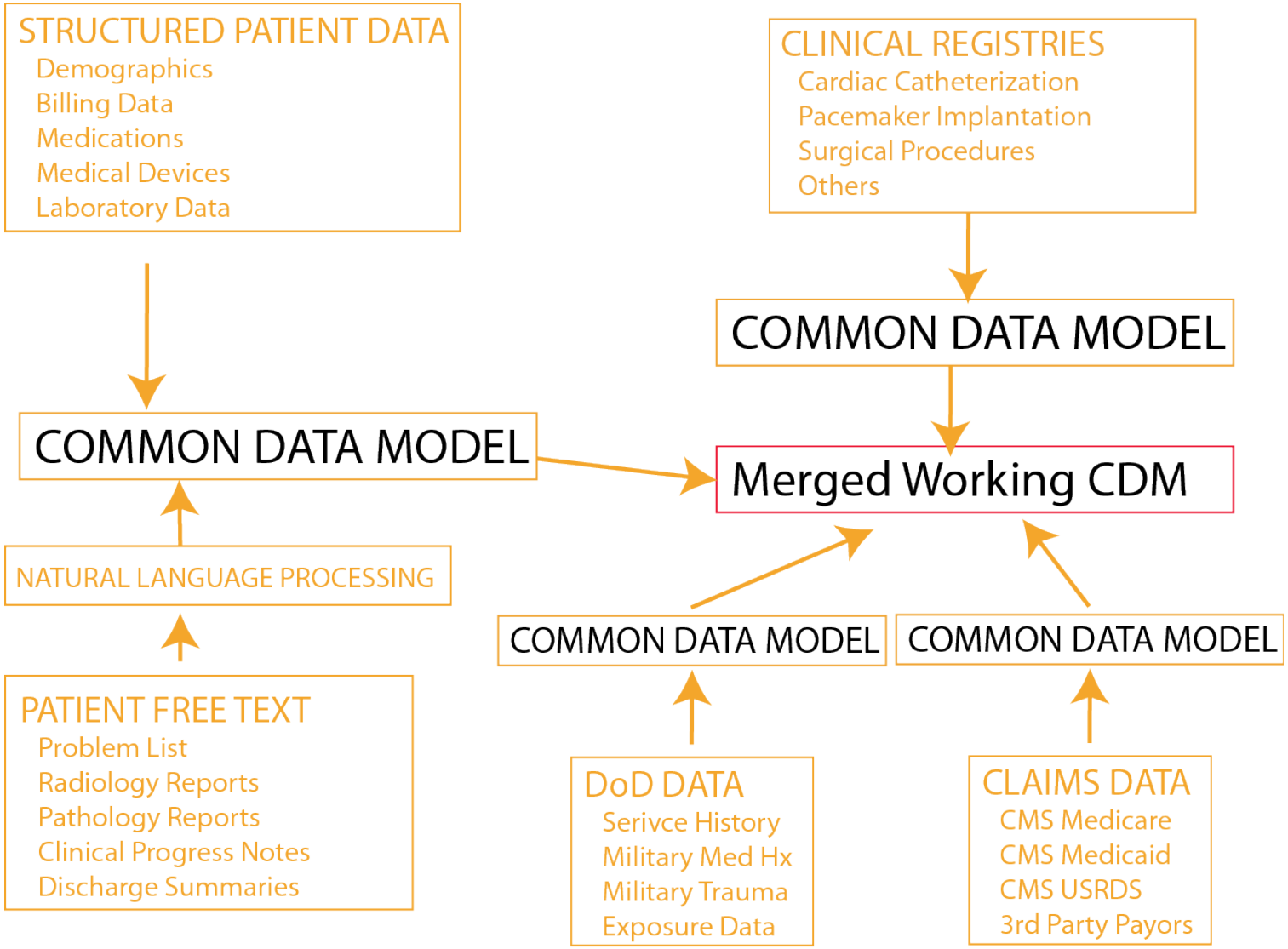
national terminologies

best practices





OMOP Strategic Roadmap



OHDSI ATLAS – Vocabulary Search

Citrix XenApp - Logged | ATLAS: Search

www.ohdsi.org/web/atlas/#/search/heart%20failure

ATLAS

- Home
- Data Sources
- Vocabulary
- Concept Sets
- Cohorts
- Incidence Rates
- Profiles
- Estimation
- Prediction
- Jobs
- Configuration
- Feedback

New Cohort Definition

Vocabulary

Search | Import

heart failure — Search

Advanced Options

Column visibility | Copy | CSV | Show 15 entries | Filter:

Showing 1 to 15 of 420 entries

Previous 1 2 3 4 5 ... 28 Next

Vocabulary
SNOMED (193)
Read (69)
LOINC (53)
ICD9CM (38)
ICD10CM (28)

Class
Clinical Finding (114)
Read (69)
Procedure (44)
Survey (42)
5-dig billing code (31)

Domain
Condition (199)
Observation (182)
Procedure (25)
Meas Value (5)
Drug (4)

Standard Concept
Standard (210)
Non-Standard (205)
Classification (5)

Invalid Reason
Valid (362)
Invalid (58)

Has Records
false (398)
true (22)

Has Descendant Records
false (390)
true (30)

Id	Code	Name	Class	RC	DRC	Domain	Vocabulary
4342296	N0000001434	Heart Failure	Ind / CI	0	2,042,725	Drug	NDFRT
500002601	500002601	OMOP Congestive Heart Failure 1	Cohort	0	267,745	Condition	Cohort
316139	84114007	Heart failure	Clinical Finding	0	262,943	Condition	SNOMED
319835	42343007	Congestive heart failure	Clinical Finding	166,433	184,923	Condition	SNOMED
443580	417996009	Systolic heart failure	Clinical Finding	6,708	48,565	Condition	SNOMED
444031	48447003	Chronic heart failure	Clinical Finding	0	44,871	Condition	SNOMED
443587	418304008	Diastolic heart failure	Clinical Finding	7,606	44,367	Condition	SNOMED
319034	60899001	Hypertensive heart disease without congestive heart failure	Clinical Finding	14,472	42,014	Condition	SNOMED
442310	56675007	Acute heart failure	Clinical Finding	0	39,101	Condition	SNOMED
40479192	441481004	Chronic systolic heart failure	Clinical Finding	10,839	30,132	Condition	SNOMED
40479576	441530006	Chronic diastolic heart failure	Clinical Finding	7,964	26,024	Condition	SNOMED
40480603	443254009	Acute systolic heart failure	Clinical Finding	6,941	25,216	Condition	SNOMED
313502	77970009	Benign hypertensive heart disease without congestive heart failure	Clinical Finding	24,163	24,163	Condition	SNOMED

OHDSI ATLAS – Phenotype/Cohort Builder

Citrix XenApp - Logged | ATLAS

www.ohdsi.org/web/atlas/#/cohortdefinition/0

ATLAS

- Home
- Data Sources
- Vocabulary
- Concept Sets
- Cohorts
- Incidence Rates
- Profiles
- Estimation
- Prediction
- Jobs
- Configuration
- Feedback

Cohort

New Cohort Definition | Save | Close

Definition | Concept Sets | Generation | Reporting | Explore | Export

Cohort definition: A cohort is defined as the set of persons satisfying one or more inclusion criteria for a duration of time. One person may qualify for one cohort multiple times during non-overlapping time intervals. Cohorts are constructed in ATLAS by specifying cohort entry criteria and cohort exit criteria. Cohort entry criteria involve selecting one or more initial events, which determine the start date for cohort entry, and optionally specifying additional inclusion criteria which filter to the qualifying events. Cohort exit criteria are applied to each cohort entry record to determine the end date when the person's episode no longer qualifies for the cohort.

All | Cohort Entry Criteria | Cohort Exit Criteria

Initial event cohort: Events are recorded time-stamped observations for the persons, such as drug exposures, conditions, procedures, measurements and visits. All events have a start date and end date, though some events may have a start date and end date with the same value (such as procedures or measurements). The event index date is set to be equal to the event start date.

People having any of the following: Add Initial Event...

with continuous observation of at least 0 days before and 0 days after event index date

Limit initial events to: earliest event per person.

Add initial event inclusion criteria

Additional qualifying inclusion criteria: The qualifying cohort will be defined as all persons who have an initial event, satisfy the initial event inclusion criteria, and fulfill all additional qualifying inclusion criteria. Each qualifying inclusion criteria will be evaluated to determine the impact of the criteria on the attrition of persons from the initial cohort.

New qualifying inclusion criteria

Please select a qualifying inclusion criteria to edit.

Limit qualifying cohort to: earliest event per person.

Cohort Exit Criteria

Cohort exit criteria: For all persons who entered the cohort, there must be a specification of when each person exits the cohort. A person must exit the cohort at the end of the observation period spanning the qualifying initial event start date, but additional cohort exit criteria may be also considered.

OMOP Direct Database / SQL Use

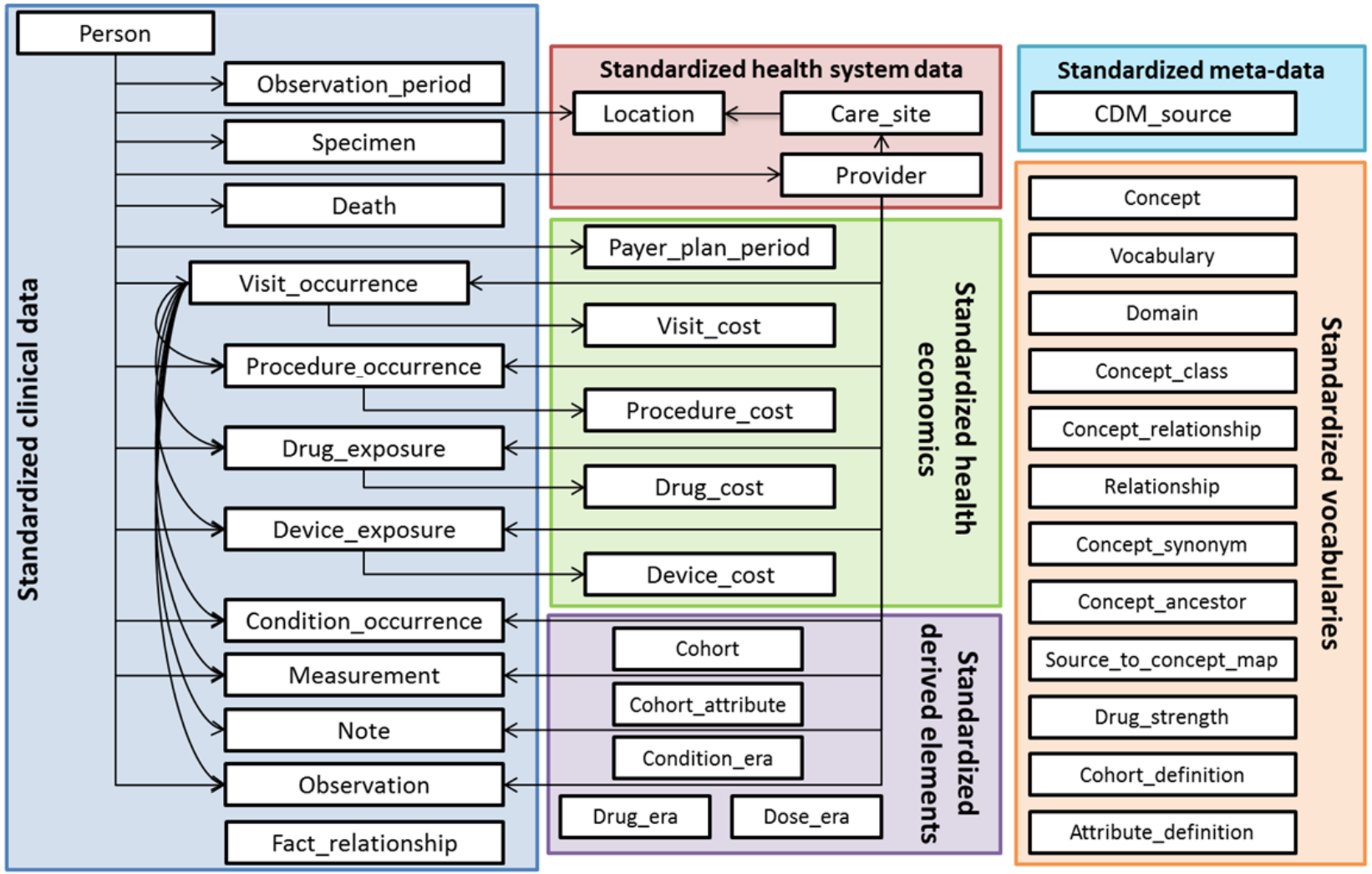
- Unfortunately, current VA software use policy has not kept up with open source software use.

- We are working towards providing future centralized ATLAS availability

- Until then, there are a number of direct database tools that approximate some ATLAS functionality, and in fact were the precursors to the ATLAS user interface functionality.

This presentation highlights ease-of-use administrative code management for OMOP database users

What does VA OMOP look like?



- Src.OMOPV5_CARE_SITE
- Src.OMOPV5_CDM_SOURCE
- Src.OMOPV5_CONCEPT
- Src.OMOPV5_CONCEPT_ANCESTOR
- Src.OMOPV5_CONCEPT_CLASS
- Src.OMOPV5_CONCEPT_RELATIONSHIP
- Src.OMOPV5_CONCEPT_SYNONYM
- Src.OMOPV5_CONDITION_ERA
- Src.OMOPV5_CONDITION_OCCURRENCE
- Src.OMOPV5_DEATH
- Src.OMOPV5_DEVICE_COST
- Src.OMOPV5_DEVICE_EXPOSURE
- Src.OMOPV5_DOMAIN
- Src.OMOPV5_DOSE_ERA
- Src.OMOPV5_DRUG_COST
- Src.OMOPV5_DRUG_ERA
- Src.OMOPV5_DRUG_EXPOSURE
- Src.OMOPV5_DRUG_STRENGTH
- Src.OMOPV5_FACT_RELATIONSHIP
- Src.OMOPV5_LOCATION
- Src.OMOPV5_MEASUREMENT
- Src.OMOPV5_NOTE
- Src.OMOPV5_OBSERVATION
- Src.OMOPV5_OBSERVATION_PERIOD
- Src.OMOPV5_PAYER_PLAN_PERIOD
- Src.OMOPV5_PERSON
- Src.OMOPV5_PROCEDURE_COST
- Src.OMOPV5_PROCEDURE_OCCURRENCE
- Src.OMOPV5_PROVIDER
- Src.OMOPV5_RELATIONSHIP
- Src.OMOPV5_SPECIMEN
- Src.OMOPV5_VISIT_COST
- Src.OMOPV5_VISIT_OCCURRENCE
- Src.OMOPV5_VOCABULARY

What does an OMOP Table look like?

OMOPV5.CONDITION_OCCURRENCE

Columns

Stock columns

- CONDITION_OCCURRENCE_ID (bigint, not null)
- PERSON_ID (bigint, null)
- CONDITION_CONCEPT_ID (int, not null)
- CONDITION_START_DATE (date, null)
- CONDITION_END_DATE (date, null)
- CONDITION_TYPE_CONCEPT_ID (int, null)
- STOP_REASON (varchar(20), null)
- PROVIDER_ID (bigint, null)
- VISIT_OCCURRENCE_ID (bigint, null)
- CONDITION_SOURCE_VALUE (varchar(50), null)
- CONDITION_SOURCE_CONCEPT_ID (int, null)

Custom columns

- x_CONDITION_START_TIME (time(7), null)
- x_CONDITION_END_TIME (time(7), null)
- x_Source_Table (varchar(50), null)
- x_Source_ID_Primary (bigint, null)
- x_Source_ID_Secondary (bigint, null)
- x_Source_ETLBatchID_Transform (int, null)
- x_ETLBatchID (int, null)
- x_DBUseStartDateTime (datetime2(0), not null)
- x_DBUseEndDateTime (datetime2(0), null)

OMOP Fact Table Links (Other Tables)

OMOPV5.CONDITION_OCCURRENCE
- Columns
  - CONDITION_OCCURRENCE_ID (bigint, not null)
  - PERSON_ID (bigint, null)
  - CONDITION_CONCEPT_ID (int, not null)
  - CONDITION_START_DATE (date, null)
  - CONDITION_END_DATE (date, null)
  - CONDITION_TYPE_CONCEPT_ID (int, null)
  - STOP_REASON (varchar(20), null)
  - PROVIDER_ID (bigint, null)
  - VISIT_OCCURRENCE_ID (bigint, null)
  - CONDITION_SOURCE_VALUE (varchar(50), null)
  - CONDITION_SOURCE_CONCEPT_ID (int, null)
  - x_CONDITION_START_TIME (time(7), null)
  - x_CONDITION_END_TIME (time(7), null)
  - x_Source_Table (varchar(50), null)
  - x_Source_ID_Primary (bigint, null)
  - x_Source_ID_Secondary (bigint, null)
  - x_Source_ETLBatchID_Transform (int, null)
  - x_ETLBatchID (int, null)
  - x_DBUseStartDateTime (datetime2(0), not null)
  - x_DBUseEndDateTime (datetime2(0), null)

Stock columns

Custom columns

Primary Key

FACT TABLE LINKS

Person Table (Demographics)

Provider Table (VA Staff)

Visit Occurrence (Encounter)

OMOP Meta-Data Links (CONCEPT Table)

OMOPV5.CONDITION_OCCURRENCE
- Columns
  - CONDITION_OCCURRENCE_ID (bigint, not null)
  - PERSON_ID (bigint, null)
  - CONDITION_CONCEPT_ID (int, not null)
  - CONDITION_START_DATE (date, null)
  - CONDITION_END_DATE (date, null)
  - CONDITION_TYPE_CONCEPT_ID (int, null)
  - STOP_REASON (varchar(20), null)
  - PROVIDER_ID (bigint, null)
  - VISIT_OCCURRENCE_ID (bigint, null)
  - CONDITION_SOURCE_VALUE (varchar(50), null)
  - CONDITION_SOURCE_CONCEPT_ID (int, null)
  - x_CONDITION_START_TIME (time(7), null)
  - x_CONDITION_END_TIME (time(7), null)
  - x_Source_Table (varchar(50), null)
  - x_Source_ID_Primary (bigint, null)
  - x_Source_ID_Secondary (bigint, null)
  - x_Source_ETLBatchID_Transform (int, null)
  - x_ETLBatchID (int, null)
  - x_DBUseStartDateTime (datetime2(0), not null)
  - x_DBUseEndDateTime (datetime2(0), null)

Stock columns

Custom columns

XXX_CONCEPT_ID

All OMOP Fact Tables have this field
Standard naming based on table
Maps to OMOP 'Standard Vocabulary' for that domain
Important

XXX_TYPE_CONCEPT_ID

This documents the type of fact put into the table
Meaning ties to CONCEPT table
Different flavors of use in different OMOP Domains
Condition & Procedure: the admin code position such as Primary/Secondary or 1st, 2nd, 3rd, 4th, etc. as well as inpatient vs outpatient

XXX_SOURCE_CONCEPT_ID

Maps to OMOP 'Source Vocabulary' from the source clinical data
Examples: ICD-9, VA Product Code, etc.

OMOP Condition Occurrence CONCEPT_IDs

[OMOPV5].[CONCEPT] Table – Example Rows

	CONCEPT_ID	CONCEPT_NAME	DOMAIN_ID	VOCABULARY_ID	CONCEPT_CODE	VALID_START_DATE	VALID_END_DATE
1	316139	Heart failure	Condition	SNOMED	84114007	1970-01-01	2099-12-31
2	444101	Hypertensive heart failure	Condition	SNOMED	46113002	1970-01-01	2099-12-31
3	44823119	Acute on chronic combined systolic and diastolic heart failure	Condition	ICD9CM	428.43	1970-01-01	2099-12-31
4	44832369	Hypertensive heart and chronic kidney disease, malignant, ...	Condition	ICD9CM	404.00	1970-01-01	2099-12-31
5	45756835	Carrier claim header - 1st position	Type Concept	Condition Type	OMOP generated	1970-01-01	2099-12-31
6	38000184	Inpatient detail - 1st position	Type Concept	Condition Type	OMOP generated	1970-01-01	2099-12-31
7	38000199	Inpatient header - primary	Type Concept	Condition Type	OMOP generated	1970-01-01	2099-12-31

CONDITION_CONCEPT_ID are the SNOMED codes

CONDITION_SOURCE_CONCEPT_ID are the ICD9CM codes

CONDITION_TYPE_CONCEPT_ID are the ‘Condition Type’ codes that dictate the type and position of the code

Source Data Reference (Back-Trace)

OMOPV5.CONDITION_OCCURRENCE

Columns

- CONDITION_OCCURRENCE_ID (bigint, not null)
- PERSON_ID (bigint, null)
- CONDITION_CONCEPT_ID (int, not null)
- CONDITION_START_DATE (date, null)
- CONDITION_END_DATE (date, null)
- CONDITION_TYPE_CONCEPT_ID (int, null)
- STOP_REASON (varchar(20), null)
- PROVIDER_ID (bigint, null)
- VISIT_OCCURRENCE_ID (bigint, null)
- CONDITION_SOURCE_VALUE (varchar(50), null)
- CONDITION_SOURCE_CONCEPT_ID (int, null)
- x_CONDITION_START_TIME (time(7), null)
- x_CONDITION_END_TIME (time(7), null)
- x_Source_Table (varchar(50), null)
- x_Source_ID_Primary (bigint, null)
- x_Source_ID_Secondary (bigint, null)
- x_Source_ETLBatchID_Transform (int, null)
- x_ETLBatchID (int, null)
- x_DBUseStartDateTime (datetime2(0), not null)
- x_DBUseEndDateTime (datetime2(0), null)

Stock columns

Custom columns

Reference to which source table fact came from

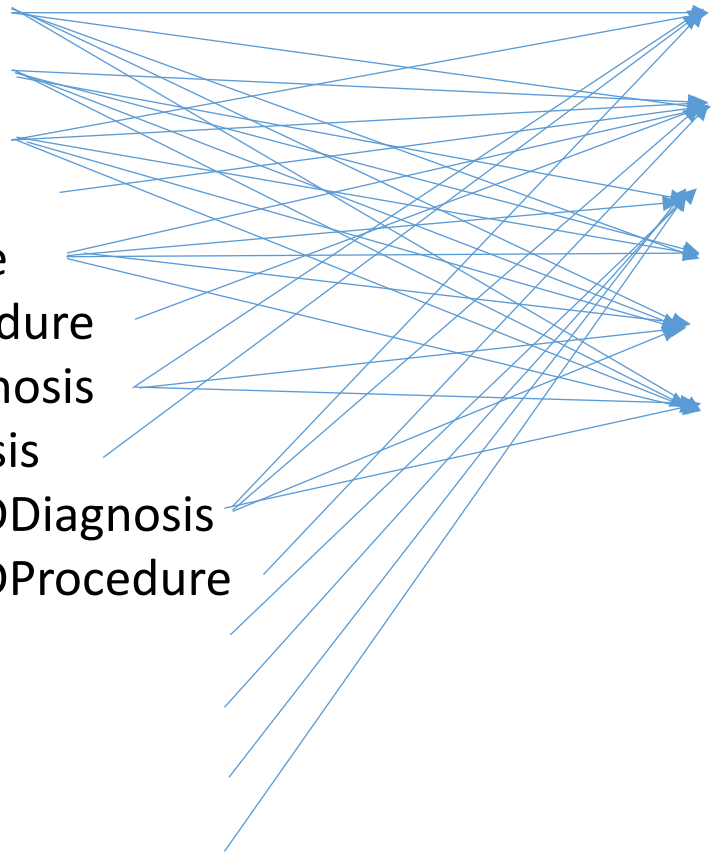
Source Data primary key identifiers for bread crumbs back to source data

ETL Track changes

Standardization of Domain Content Location

Source Tables

Outpat.VDiagnosis
Outpat.Vprocedure
InpatientDiagnosis
InpatientICDProcedure
InpatientCPTProcedure
InpatientSurgicalProcedure
InpatientFeeBasisDiagnosis
PatientTransferDiagnosis
FeeInpatientInvoiceICDDiagnosis
FeeInpatientInvoiceICDProcedure
RxOutPatFill
BCMADispensed
BCMASolution
BCMAAdditive



Destination Tables

Condition_Occurrence
Procedure_Occurrence
Drug_Exposure
Device_Exposure
Measurement
Observation

Table Value Functions

Script Library

- Developed by user community for OMOP v5 before ATLAS Release
- Adapted for V5 by VINCI and installed on CDWWork in rb02/03
- Only meta-data, no patient data
- Very helpful support scripts for all OMOP domains
- Approx. 50 support functions
- Detailed documentation in VA pulse
- General use parallels examples given today

Table Value Functions In This Example

TVF_V01_TranslateSourceCodeToConditionConcepts

This function shows the SOURCE_CONCEPT_ID for any code along with the target code & CONCEPT_ID

TVF_V02_SourceCodeMappingList

This function collapses the SOURCE and TARGET codes and CONCEPT_ID's into a list suitable for use in SQL filtering joins.

PARAMETERS:

SOURCE_CODE

This is the source vocabulary code
Example : 250.11

SOURCE_VOCABULARY

This is the vocabulary id from [VOCABULARY]
Example: ICD9CM

INDEX_DATE

This is the restriction date for whether a code is active. NULL does not enforce any active filter.

OMOPV5.VOCABLARY Table

[OMOPV5].[VOCABULARY] Excerpted Rows For This Example

	VOCABULARY_ID	VOCABULARY_NAME	VOCABULARY_VERSION	VOCABULARY_CONCEPT_ID
1	CPT4	Current Procedural Terminology version 4 (AMA)	2017AA	44819100
2	HCPCS	Healthcare Common Procedure Coding System (CMS)	2016 Alpha Numeric HCPCS File	44819101
3	ICD10	International Classification of Diseases, Tenth Revision (WHO)	2016 Release	44819124
4	ICD10CM	International Classification of Diseases, Tenth Revision, Clinical Modificat...	ICD10CM FY2017 code descriptions	44819098
5	ICD10PCS	ICD-10 Procedure Coding System (CMS)	ICD10PCS 20160518	44819125
6	ICD9CM	International Classification of Diseases, Ninth Revision, Clinical Modificati...	ICD9CM v32 master descriptions	5046
7	ICD9Proc	International Classification of Diseases, Ninth Revision, Clinical Modificati...	ICD9CM v32 master descriptions	44819099
8	NDC	National Drug Code (FDA and manufacturers)	dm+d 20150817	44819105
9	SNOMED	Systematic Nomenclature of Medicine - Clinical Terms (IHTSDO)	SnomedCT Release 20170401	44819097
10	VA Product	VA National Drug File Product (VA)	RxNorm Full 20170807	44819120

Table Value Functions

TVF_V01_TranslateSourceCodeToConditionConcepts

```
The following example is how to use a TVF script to retreive the OMOP concept ID for a single ICD-9 Code, along with
the translation of that code to SNOMED CONCEPT ID(s) or other target vocabulary (works for all domains):
SELECT * FROM [OMOPV5].[tvf_V01_TranslateSourceCodeToConditionConcepts] ('250.01','ICD9CM',null)
```

100 %

Results | Messages

	Source_code	Source_concept_id	Source_Concept_Name	Source_Domain_ID	Source_Vocab_Name	Mapping...	Target_Co...	Target_concept_id	Target_Concept_Name	Target_Domain_ID	Target_Vocab_N
1	250.01	44820682	Diabetes mellitus without mention of complicatio...	Condition	International Classific...	Maps to	46635009	201254	Type 1 diabetes melli...	Condition	Systematic Nome

TVF_V02_SourceCodeMappingList

```
The following example is how to use the TVF script to get all OMOP CONCEPT_ID matches for a single ICD-9 code, this is
the core useful TVF that helps users build CONCEPT_ID filter lists for their needs:
SELECT * FROM [OMOPV5].[tvf_V02_SoureCodeMappingList] ('250.11','ICD9CM',null)
```

100 %

Results | Messages

	Concept_code	Concept_id	Concept_Name	Domain_ID	Vocab_Name
1	250.11	44824071	Diabetes with ketoacidosis, type I [juvenile type], not stated as uncontrolled	Condition	International Classification of Diseases, Ninth Revision, Clinical Modifi...
2	420270002	439770	Ketoacidosis in type 1 diabetes mellitus	Condition	Systematic Nomenclature of Medicine - Clinical Terms (IHTSDO)

Another Way– OMOPV5Dim Tables

- If you are interested in quickly assessing whether codes are used in the VA and what they may be mapped to, you can also use the OMOPV5Dim tables:
  - OMOPV5Dim.ICD9_CONCEPT
  - OMOPV5Dim.ICD10_CONCEPT
  - OMOPV5Dim.ICD9Procedure_CONCEPT
  - OMOPV5Dim.ICD10Procedure_CONCEPT
  - OMOPV5Dim.CPT
  - OMOPV5Dim.LocalDrug_CONCEPT
- This also includes a global count of the number of instances of a source code in the source data

Covered in prior talks and user group meetings

Create a Temporary or Permanent Code List Table

```
CREATE table #TempList (
        [Source_Code] [varchar](50) null
        ,[Vocabulary_ID] [varchar](50) null
        ,[DateFilter] [date] null )
```

Insert all the Heart Failure Codes into the table

	Source_Code	Vocabulary_ID	DateFilter
1	428	ICD9CM	NULL
2	428	ICD9CM	NULL
3	428.0	ICD9CM	NULL
4	428.1	ICD9CM	NULL
5	428.2	ICD9CM	NULL
6	428.20	ICD9CM	NULL
7	428.21	ICD9CM	NULL
8	428.22	ICD9CM	NULL
9	428.23	ICD9CM	NULL

Use CROSS APPLY to generate a CONCEPT_ID list from a Source Code list

```
select *
INTO #TempFilter
from #TempList [X]
  cross apply [OMOPV5].[tvf_V02_SoureCodeMappingList] ([X].[Source_Code],[X].[Vocabulary_ID],[X].[DateFilter]);
create clustered index [ind_123] on #TempFilter ([CONCEPT_ID]);
```

OMOP Table Filter (TVF Cross Apply Output)

	CONCEPT_CODE	CONCEPT_ID	CONCEPT_NAME	DOMAIN_ID	VOCAB_NAME
22	443253003	40480602	Acute on chronic systolic heart failure	Condition	Systematic Nomenclature of Medicine - Clinical Terms (IHTSDO)
23	443254009	40480603	Acute systolic heart failure	Condition	Systematic Nomenclature of Medicine - Clinical Terms (IHTSDO)
24	443343001	40481042	Acute diastolic heart failure	Condition	Systematic Nomenclature of Medicine - Clinical Terms (IHTSDO)
25	443344007	40481043	Acute on chronic diastolic heart failure	Condition	Systematic Nomenclature of Medicine - Clinical Terms (IHTSDO)
26	442304009	40482727	Combined systolic and diastolic dysfunction	Condition	Systematic Nomenclature of Medicine - Clinical Terms (IHTSDO)
27	442304009	40482727	Combined systolic and diastolic dysfunction	Condition	Systematic Nomenclature of Medicine - Clinical Terms (IHTSDO)
28	404.01	44819692	Hypertensive heart and chronic kidney disease, malignant, with ...	Condition	International Classification of Diseases, Ninth Revision, Clinical Modification, ...
29	404.03	44819693	Hypertensive heart and chronic kidney disease, malignant, with ...	Condition	International Classification of Diseases, Ninth Revision, Clinical Modification, ...
30	404.11	44819695	Hypertensive heart and chronic kidney disease, benign, with he...	Condition	International Classification of Diseases, Ninth Revision, Clinical Modification, ...
31	404.91	44819696	Hypertensive heart and chronic kidney disease, unspecified, wit...	Condition	International Classification of Diseases, Ninth Revision, Clinical Modification, ...
32	404.13	44820856	Hypertensive heart and chronic kidney disease, benign, with he...	Condition	International Classification of Diseases, Ninth Revision, Clinical Modification, ...

Combines Source and Target Vocabulary CONCEPT_ID’s
-> agnostic to join on CONCEPT_ID or SOURCE_CONCEPT_ID

Find All Patients with HF Codes of Interest

```
SELECT [A].[PERSON_ID]
    ,[CONDITION_CONCEPT_ID
            ,[CONDITION_TYPE_CONCEPT_ID]
            ,CASE WHEN [CONDITION_TYPE_CONCEPT_ID] in (38000183,38000184,38000199,38000200
                                                       ,44786627,44786628,45756835,45756843)
                    THEN 1 ELSE 0 END as [PrimaryCode]
            ,CASE WHEN [Visit].[VISIT_CONCEPT_ID] in (9201,800000001) THEN 1 ELSE 0 END as [InpatientCode]
           ,[CONDITION_START_DATE]
           ,[A].[VISIT_OCCURRENCE_ID]
INTO #TempCondition
FROM [OMOPV5].[CONDITION_OCCURRENCE] a
INNER JOIN #TempFilter b
on a.[CONDITION_CONCEPT_ID] = b.concept_id
left join [OMOPV5].[VISIT_OCCURRENCE] [Visit]
on [A].[VISIT_OCCURRENCE_ID] = [Visit].[VISIT_OCCURRENCE_ID]
```

Find All Patients with HF Codes
SQL Code Anatomy

```
SELECT
        [A].[PERSON_ID]
        ,[CONDITION_CONCEPT_ID
        ,[CONDITION_START_DATE]
        ,[A].[VISIT_OCCURRENCE_ID]

FROM [OMOPV5].[CONDITION_OCCURRENCE] a
INNER JOIN #TempFilter b
on a.[CONDITION_CONCEPT_ID] = b.concept_id
left join [OMOPV5].[VISIT_OCCURRENCE] [Visit]
on [A].[VISIT_OCCURRENCE_ID] = [Visit].[VISIT_OCCURRENCE_ID]
```

Find All Patients with HF Codes
SQL Code Anatomy

```
,CASE WHEN [CONDITION_TYPE_CONCEPT_ID] in (38000183,38000184,38000199,38000200
                                          ,44786627,44786628,45756835,45756843)
THEN 1 ELSE 0 END as [PrimaryCode]
```

How do you know which codes are the Primary or 1st Position Diagnosis Codes? Where do you find them?

SELECT * FROM [OMOPV5].[CONCEPT] WHERE [VOCABULARY_ID] = 'Condition Type'

	CONCEPT_ID	CONCEPT_NAME	DOMAIN_ID	VOCABULARY_ID	STANDARD_CONCEPT
1	38000183	Inpatient detail - primary	Type Concept	Condition Type	S
2	38000184	Inpatient detail - 1st position	Type Concept	Condition Type	S
3	38000199	Inpatient header - primary	Type Concept	Condition Type	S
4	38000200	Inpatient header - 1st position	Type Concept	Condition Type	S
5	44786627	Primary Condition	Type Concept	Condition Type	S
6	44786628	First Position Condition	Type Concept	Condition Type	S
7	45756835	Carrier claim header - 1st position	Type Concept	Condition Type	S
8	45756843	Carrier claim detail - 1st position	Type Concept	Condition Type	S

(Rows Reflect only SQL Case Statement)

Find All Patients with HF Codes
SQL Code Anatomy

```
,CASE WHEN [Visit].[VISIT_CONCEPT_ID] in (9201,800000001) THEN 1 ELSE 0 END as [InpatientCode]
```

How do you know which codes are the inpatient and outpatient visit types? Where do you find them?

Same Answer!

```
SELECT * FROM [OMOPV5].[CONCEPT] WHERE [VOCABULARY_ID] = 'Visit'
```

	CONCEPT_ID	CONCEPT_NAME	DOMAIN_ID	VOCABULARY_ID	STANDARD_CONCEPT
1	42898160	Long Term Care Visit	Visit	Visit	S
2	9203	Emergency Room Visit	Visit	Visit	S
3	9202	Outpatient Visit	Visit	Visit	S
4	9201	Inpatient Visit	Visit	Visit	S
5	262	Emergency Room and Inpatient Visit	Visit	Visit	S
6	800000001	Inpatient Observation Visit	Visit	Visit	S
7	800000002	Outpatient Visit Within Inpatient Visit	Visit	Visit	S

Logic (Pseudo-Code) for Date Calculation

- If Primary Inpatient Code, Heart Failure start date is 1st code instance
- Otherwise (Inpatient Secondary or any Outpatient), start date is 2nd code instance.
- Collapse different category of codes into single instance per date so a patient can't qualify by multiple codes in the same day
- No stopping date of heart failure other than data truncation date

- All Code Provided in Example SQL Script

Example of Table Output – Synthetic Data

	PERSON_ID	InpatientPrimary	CONDITION_START_DATE	QUALIFYING_START_DATE
1	1	0	2003-07-22	NULL
2	2	1	2016-02-05	2016-02-05
3	3	0	2016-01-08	NULL
4	4	0	2002-04-04	2003-04-14
5	4	0	2003-04-14	2003-04-14
6	4	0	2003-06-17	2003-04-14
7	4	0	2003-10-15	2003-04-14
8	4	0	2004-02-12	2003-04-14
9	4	0	2004-04-13	2003-04-14
10	5	0	2010-05-10	NULL
11	6	0	2014-04-13	NULL
12	7	1	2016-04-14	2016-04-14
13	8	0	2007-11-08	NULL
14	9	0	2008-04-14	2008-07-15
15	9	0	2008-07-15	2008-07-15
16	9	0	2008-12-02	2008-07-15

Persistent Storage into OMOP COHORT Table

```sql
INSERT INTO [OMOPV5].[COHORT] (
    [COHORT_DEFINITION_ID]
    ,[SUBJECT_ID]
    ,[COHORT_START_DATE]
    ,[COHORT_END_DATE]
    )
SELECT
    2017110701 as [COHORT_DEFINITION_ID]
    ,[PERSON_ID] as [SUBJECT_ID]
    ,[QUALIFYING_START_DATE] as [COHORT_START_DATE]
    ,'11/07/2017' as [COHORT_END_DATE]
FROM (
    SELECT distinct [person_Id],[qualifying_start_date]
    from omop_v5_qa.temp.temppresent
    where [qualifying_start_date] is not null
    ) x
```

100 %

Results | Messages

	COHORT_DEFINITION_ID	SUBJECT_ID	COHORT_START_DATE	COHORT_END_DATE
1	2017110701	2	2016-02-05	2017-11-07
2	2017110701	4	2003-04-14	2017-11-07
3	2017110701	7	2016-04-14	2017-11-07
4	2017110701	9	2008-07-15	2017-11-07

```sql
INSERT INTO [OMOPV5].[COHORT_DEFINITION] (
    [COHORT_DEFINITION_ID]
    ,[COHORT_DEFINITION_NAME]
    ,[COHORT_DEFINITION_DESCRIPTION]
    ,[DEFINITION_TYPE_CONCEPT_ID]
    ,[COHORT_DEFINITION_SYNTAX]
    ,[SUBJECT_CONCEPT_ID]
    ,[COHORT_INITIATION_DATE]
    )
VALUES (
    2017110701  /* [COHORT_DEFINITION_ID] */
    ,'Heart Failure (Modified Go 2006)' /* [COHORT_DEFINITION_NAME] */
    ,'   Based on Go, et al. Circulation 2006;113(23):2713-2723 that used combo inpatient/outpatient codes
        with >= 1 hospitalization with CHF as primary code
        or >= 2 any codes (inpatient/ED/outpatient primary secondary)
        PPV 97% compared to physician review
        Original Publication Codes: 398.91, 402.01, 402.11, 402.91, 428.0, 428.1, or 428.9
        Implemented Codes:  See COHORT_DEFINITION_SYNTAX, using additional codes from Mini-Sentinel HF Review
        ' /*  [COHORT_DEFINITION_DESCRIPTION] */
        ,-1 /* DEFINITION_TYPE_CONCEPT_ID */  /* future work */
        ,'<INSERT PROGRAMMING CODE>' /* [COHORT_DEFINITION_SYNTAX] */
        ,56 /* [SUBJECT_CONCEPT_ID] */ /* this is the meta-data concept for PERSON, defines the domain of SUBJECT_ID in COHORT
        ,'11/07/2017' /* [COHORT_INITIATION_DATE] */
    )
```

100 %

Results | Messages

	COHORT_DEFINITION_ID	COHORT_DEFINITION_NAME	COHORT_DEFINITION_DESCRIPTION	SUBJECT_CONCEPT_ID	COHORT_INITIATION_DATE
1	2017110701	Heart Failure (Modified Go 2006)	Based on Go, et al. Circulation 2006;113(23):...	56	2017-11-07

Total Programming Lines of Code:

Excluding Comments

- Create Table of ICD-9 Codes for Filter: 34
- Create Basic Computable Phenotype: 66
- Load OMOP COHORT & COHORT DEF. Tables for Re-Use: ~30

More complex logic will dictate more complex code, but….

- Each Phenotype can be developed independently with re-usable code across health care systems,
- If deployed in OMOP CDM, re-usable across research projects at very low computational and human programming cost
- Same code for CDW Only, or CDW/Medicare Merged, or CDW/Medicare/DoD Merged…

Administrative Code Summary

- The structure of OMOP allows for a wide variety of source data tables to be aggregated into a small number of standardized tables
- Mapping and mapping relationships help support vocabulary cross-walks
- Computable phenotype building, documentation, and persistent storage are supported in OMOP
- Easy use of OMOP relies on graphical user tools (community) or short-cut SQL scripts (abstracts need to understand all of OMOP table inter-dependencies)

Transition to Medications

- Not enough time in this presentation for a full use case example
- Highlight how medications are mapped in VA OMOP
- Highlight a few great TVF's that are used to roll up medications to a drug class
- Future Presentation: using medications with a full use case

Drug Exposure Table

The anatomy of this table is similar to Condition Occurrence

Some unique fields related to medications:

- Stop Reason
- Refills
- Quantity
- Days Supply
- Route
- Dose

But for this overview we will focus only on the same concepts as befor

- DRUG_CONCEPT_ID
- DRUG_TYPE_CONCEPT_ID
- DRUG_SOURCE_CONCEPT_ID

OMOPV5.DRUG_EXPOSURE

Columns

- DRUG_EXPOSURE_ID (bigint, not null)
- PERSON_ID (bigint, null)
- DRUG_CONCEPT_ID (bigint, null)
- DRUG_EXPOSURE_START_DATE (date, null)
- DRUG_EXPOSURE_END_DATE (date, null)
- DRUG_TYPE_CONCEPT_ID (int, not null)
- STOP_REASON (varchar(20), null)
- REFILLS (numeric(3,0), null)
- QUANTITY (numeric(4,0), null)
- DAYS_SUPPLY (numeric(4,0), null)
- SIG (varchar(500), null)
- ROUTE_CONCEPT_ID (int, null)
- EFFECTIVE_DRUG_DOSE (float, null)
- DOSE_UNIT_CONCEPT_ID (int, null)
- LOT_NUMBER (varchar(50), null)
- PROVIDER_ID (bigint, null)
- VISIT_OCCURRENCE_ID (bigint, null)
- DRUG_SOURCE_VALUE (varchar(8000), null)
- DRUG_SOURCE_CONCEPT_ID (int, null)
- ROUTE_SOURCE_VALUE (varchar(50), null)
- DOSE_UNIT_SOURCE_VALUE (varchar(50), null)
- DRUG_EXPOSURE_START_DATETIME (datetime2(0), null)
- DRUG_EXPOSURE_END_DATETIME (datetime2(0), null)
- x_LocalDrugSID (int, null)
- x_CARE_SITE_ID (int, null)
- x_Source_Table (nvarchar(50), null)
- x_Source_ID_Primary (bigint, null)
- x_Source_ID_Secondary (bigint, null)
- x_Source_ETLBatchID_Transform (int, null)
- x_ETLBatchID (int, null)
- x_DBUseStartDateTime (datetime2(0), not null)
- x_DBUseEndDateTime (datetime2(0), null)
- x_DimLocalDrugMap_ETLBatchID (int, null)

DRUG_TYPE_CONCEPT_ID

```
USE [OMOP_V5]
GO

select * from omop_v5.omopv5.concept
where vocabulary_id = 'Drug Type'
```

100 %

Results | Messages

	CONCEPT_ID	CONCEPT_NAME	DOMAIN_ID	VOCABULARY_ID	CONCEPT_CLASS_ID	STANDARD_CONCEPT	CONCEPT
1	581373	Physician administered drug (identified from EHR...	Type Concept	Drug Type	Drug Type	S	OMOP ge
2	38000175	Prescription dispensed in pharmacy	Type Concept	Drug Type	Drug Type	S	OMOP ge
3	38000176	Prescription dispensed through mail order	Type Concept	Drug Type	Drug Type	S	OMOP ge
4	38000177	Prescription written	Type Concept	Drug Type	Drug Type	S	OMOP ge
5	38000178	Medication list entry	Type Concept	Drug Type	Drug Type	S	OMOP ge
6	38000179	Physician administered drug (identified as proced...	Type Concept	Drug Type	Drug Type	S	OMOP ge
7	38000180	Inpatient administration	Type Concept	Drug Type	Drug Type	S	OMOP ge
8	38000181	Drug era - 0 days persistence window	Type Concept	Drug Type	Drug Type	S	OMOP ge
9	38000182	Drug era - 30 days persistence window	Type Concept	Drug Type	Drug Type	S	OMOP ge
10	43542356	Physician administered drug (identified from EHR...	Type Concept	Drug Type	Drug Type	S	OMOP ge
11	43542357	Physician administered drug (identified from referr...	Type Concept	Drug Type	Drug Type	S	OMOP ge
12	43542358	Physician administered drug (identified from EHR...	Type Concept	Drug Type	Drug Type	S	OMOP ge
13	44777970	Randomized Drug	Type Concept	Drug Type	Drug Type	S	OMOP ge
14	44787730	Patient Self-Reported Medication	Type Concept	Drug Type	Drug Type	S	OMOP ge

Medication Summary

- Medication navigation and roll-up/aggregation is easy with the OMOP (UMLS) meta-data
- Relies on LocalDrugSID -> RxNorm mapping, and this is a work in progress
- Future discussions:
  - Drug Eras
  - Dose, Route, etc

VA Medications: Based on NDC & VA Product

SOURCE_CONCEPT_ID = VA Product (VUID) and NDC Mappings

CONCEPT_ID = RxNorm Mappings

OMOPV5Dim.LocalDrug_CONCEPT –

Contains all the maps of the LocalDrugSID's from CDW
Processed to find the NDC and VUID values
Mapped through to RxNorm target vocabulary

	LocalDrugSID	LocalDrugNameWithDose	NationalDrugSID	NDC	VUID	CONCEPT_ID	CONCEPT_NAME	DOMAIN_ID	SOURCE_CONCEPT_ID	Instance_Count
1	580059	METOPROLOL TARTRATE 50MG TAB	202447	52343-0060-99	4004608	40167218	Metoprolol Tartrate 50 MG Oral Tablet	Drug	42567529	1108960
2	12384	METOPROLOL TARTRATE 50MG TAB	264231	52343-0060-99	4004608	40167218	Metoprolol Tartrate 50 MG Oral Tablet	Drug	42567529	1091918
3	693509	METOPROLOL TARTRATE 50MG TAB	1345037	52343-0060-99	4004608	40167218	Metoprolol Tartrate 50 MG Oral Tablet	Drug	42567529	1013248
4	354793	METOPROLOL TARTRATE 50MG TAB	1371484	52343-0060-99	4004608	40167218	Metoprolol Tartrate 50 MG Oral Tablet	Drug	42567529	856864
5	220321	METOPROLOL TARTRATE 25MG TAB	148699	00378-0018-01	4016944	40167213	Metoprolol Tartrate 25 MG Oral Tablet	Drug	42569833	744655

Building a Medication Drug Class Filter List

- A straight forward way to build a drug list filter by following the UMLS RxNorm, ATC, and VA Product class relationships, which are all in UMLS and thus in the OMOP relationship table

Find a Drug CONCEPT_ID for your drug

```
SELECT top 5 * FROM [OMOPV5].[tvf_D02_GetDrugOrDrugClassByKeyword] ( 'metoprolol',null)
GO
```

100 %

Results | Messages

	Entity_concept_Id	Entity_name	Entity_code	Entity_type	Entity_concept_class_id	Entity_vocabulary_id	Entity_vocabulary_name
1	19000867	Metoprolol fumarate	102313	Concept	Precise Ingredient	RxNorm	RxNorm (NLM)
2	19000868	Metoprolol 190 MG Oral Tablet [Metoros]	102314	Concept	Branded Drug	RxNorm	RxNorm (NLM)
3	19000869	Metoprolol 95 MG Oral Tablet [Metoros]	102315	Concept	Branded Drug	RxNorm	RxNorm (NLM)
4	19003704	Metoprolol 200 MG Extended Release Tablet [Betal...	104312	Concept	Branded Drug	RxNorm	RxNorm (NLM)
5	19003705	Metoprolol 200 MG 24 Hour Extended Release Tabl...	104313	Concept	Branded Drug	RxNorm	RxNorm (NLM)

Randomly select one of the CONCEPT_ID's for a target drug from TVF list

Get RxNorm Drug Ingredient for Any Drug

```
SELECT * FROM [OMOPV5].[tvf_D03_GetDrugIngredientsByID] (19000869,null)
GO
```

100 %

Results | Messages

	Drug_concept_id	Drug_name	Drug_concept_code	Drug_concept_class_ID	Ingredient_concept_id	Ingredient_name	Ingredient_concept_code	Ingredient_concept_class_ID
1	19000869	Metoprolol 95 MG Oral Tablet [Metoros]	102315	Branded Drug	1307046	Metoprolol	6918	Ingredient

Determine the drug ingredient CONCEPT_ID(s) for the drug of interest

Find the Drug Classes (Ancestors) of a Drug Ingredient

```
SELECT * FROM [OMOPV5].[tvf_D08_GetDrugClassesForDrugOrIngredient] (1307046,null)
```

100 %

Results | Messages

	Class_concept_id	Class_name	Class_code	Classification_id	Class_vocabulary_id	Class_vocabulary_name	Levels_of_separation
1	4279486	CARDIOVASCULAR MEDICATIONS	CV000	VA Class	VA Class	VA National Drug File Class (VA)	2
2	4279487	BETA BLOCKERS/RELATED	CV100	VA Class	VA Class	VA National Drug File Class (VA)	0
3	4283993	ANTIHYPERTENSIVE COMBINATIONS	CV400	VA Class	VA Class	VA National Drug File Class (VA)	0
4	4253999	Phenoxypropanolamines	N0000166515	Chemical Structure	NDFRT	National Drug File - Reference Terminology (VA)	3
5	4324566	G-Protein-linked Receptor Interactions	N0000000152	Mechanism of Action	NDFRT	National Drug File - Reference Terminology (VA)	5
6	4324568	Adrenergic beta-Antagonists	N0000000161	Mechanism of Action	NDFRT	National Drug File - Reference Terminology (VA)	1
7	4324585	Receptor Interactions	N0000000085	Mechanism of Action	NDFRT	National Drug File - Reference Terminology (VA)	7
8	4324595	Adrenergic Receptor Interactions	N0000000153	Mechanism of Action	NDFRT	National Drug File - Reference Terminology (VA)	4
9	4324988	Adrenergic Antagonists	N0000000092	Mechanism of Action	NDFRT	National Drug File - Reference Terminology (VA)	2
10	4325133	Cellular or Molecular Interactions	N0000000223	Mechanism of Action	NDFRT	National Drug File - Reference Terminology (VA)	9
11	4327233	Cardiovascular Activity Alteration	N0000008331	Physiologic Effect	NDFRT	National Drug File - Reference Terminology (VA)	7
12	4327525	Cardiac Rate Alteration	N0000008329	Physiologic Effect	NDFRT	National Drug File - Reference Terminology (VA)	1
13	4331195	Negative Chronotropy	N0000009756	Physiologic Effect	NDFRT	National Drug File - Reference Terminology (VA)	0
14	4331196	Negative Inotropy	N0000009757	Physiologic Effect	NDFRT	National Drug File - Reference Terminology (VA)	0
15	4331199	Organ System Specific Effects	N0000009770	Physiologic Effect	NDFRT	National Drug File - Reference Terminology (VA)	8
16	4331211	Physiological Effects	N0000009802	Physiologic Effect	NDFRT	National Drug File - Reference Terminology (VA)	9
17	4333201	Adrenergic beta1-Antagonists	N0000009923	Mechanism of Action	NDFRT	National Drug File - Reference Terminology (VA)	0
18	4333329	Cardiac Contractility Alteration	N0000008328	Physiologic Effect	NDFRT	National Drug File - Reference Terminology (VA)	1
19	4340569	Chemical Ingredients	N0000000002	Chemical Structure	NDFRT	National Drug File - Reference Terminology (VA)	11

If you already know the VA Drug Class…

```sql
SELECT [CONCEPT_ID],[CONCEPT_NAME],[DOMAIN_ID],[VOCABULARY_ID],[CONCEPT_CLASS_ID],[STANDARD_CONCEPT],[CONCEPT_CODE]
  FROM [OMOP_V5].[OMOPV5].[CONCEPT]
  where concept_name like '%beta%'
  and vocabulary_id = 'VA Class'
```

100 %

Results | Messages

	CONCEPT_ID	CONCEPT_NAME	DOMAIN_ID	VOCABULARY_ID	CONCEPT_CLASS_ID	STANDARD_CONCEPT	CONCEPT_CODE
1	4279200	BETA-LACTAMS ANTIMICROBIALS,OTHER	Drug	VA Class	VA Class	C	AM119
2	4279356	PENICILLINS AND BETA-LACTAM ANTIMICROBIALS	Drug	VA Class	VA Class	C	AM114
3	4279487	BETA BLOCKERS/RELATED	Drug	VA Class	VA Class	C	CV100
4	4280126	BETA-BLOCKERS,TOPICAL OPHTHALMIC	Drug	VA Class	VA Class	C	OP101
5	4280413	BETA-BLOCKERS,SYSTEMIC OPHTHALMIC	Drug	VA Class	VA Class	C	OP107

Determine all Drug Ingredients for Class

```
SELECT * FROM [OMOPV5].[tvf_D10_GetIngredientByDrugClass] (4279487,null)
```

100 %

Results | Messages

	ingredient_concept_id	ingredient_concept_name	ingredient_concept_class_id	ingredient_concept_code
1	42629333	Propranolol Injection	Clinical Drug Form	1789973
2	42629334	Propranolol Hydrochloride 1 MG/ML Injection	Clinical Drug	1789975
3	43014173	metoprolol succinate 50 MG	Clinical Drug Comp	1370474
4	43014171	metoprolol succinate 25 MG	Clinical Drug Comp	1370489
5	43014169	metoprolol succinate 200 MG	Clinical Drug Comp	1370500
6	43014167	metoprolol succinate 100 MG	Clinical Drug Comp	1370483
7	42707472	4 ML Labetalol hydrochloride 5 MG/ML Cartridge	Quant Clinical Drug	1234256
8	45775831	Sotalol Hydrochloride 5 MG/ML Oral Solution	Clinical Drug	1593725
9	45775835	Sotalol Hydrochloride 5 MG/ML Oral Solution [Soty...	Branded Drug	1593731
10	46221721	Metoprolol Tartrate 37.5 MG	Clinical Drug Comp	1606346
11	46221722	Metoprolol Tartrate 37.5 MG Oral Tablet	Clinical Drug	1606347
12	46221723	Metoprolol Tartrate 75 MG	Clinical Drug Comp	1606348
13	46221724	Metoprolol Tartrate 75 MG Oral Tablet	Clinical Drug	1606349
14	45775830	Sotalol Oral Solution	Clinical Drug Form	1593724
15	45775829	Sotalol Hydrochloride 5 MG/ML	Clinical Drug Comp	1593722
16	44816220	Propranolol Hydrochloride 4.28 MG/ML Oral Soluti...	Branded Drug	1495064
17	44816219	Propranolol Hydrochloride 4.28 MG/ML Oral Solution	Clinical Drug	1495058
18	44815867	Propranolol Hydrochloride 4.28 MG/ML	Clinical Drug Comp	1495057
19	42800544	Betaxolol Hydrochloride 10 MG	Clinical Drug Comp	1297752

338 ingredient concepts

Expand Drug Ingredients to All Formulations

```
SELECT * FROM [OMOPV5].[tvf_D04_GetDrugsByIngredientID] (1307046,null)
```

100 %

Results | Messages

	Drug_concept_id	Drug_Vocabulary_ID	Drug_name	Drug_concept_code	Drug_concept_class_id
1	974431	RxNorm	Hydrochlorothiazide 12.5 MG / Metoprolol 100 MG ...	246267	Clinical Drug
2	1307046	RxNorm	Metoprolol	6918	Ingredient
3	1307122	RxNorm	Metoprolol 200 MG	323895	Clinical Drug Comp
4	1307123	RxNorm	Metoprolol 50 MG	328469	Clinical Drug Comp
5	1307125	RxNorm	Metoprolol 25 MG	331561	Clinical Drug Comp
6	19000868	RxNorm	Metoprolol 190 MG Oral Tablet [Metoros]	102314	Branded Drug
7	19000869	RxNorm	Metoprolol 95 MG Oral Tablet [Metoros]	102315	Branded Drug
8	19003734	RxNorm	Chlorthalidone 12.5 MG / Metoprolol 100 MG Oral T...	104332	Clinical Drug
9	19011442	RxNorm	Metoprolol 190 MG Oral Tablet	141833	Clinical Drug
10	19011443	RxNorm	Metoprolol 95 MG Oral Tablet	141834	Clinical Drug
11	19025091	RxNorm	Metoprolol Tartrate 50 MG Oral Tablet [Betaloc]	201331	Branded Drug
12	19025092	RxNorm	Metoprolol Tartrate 50 MG Oral Tablet [Mepranix]	201332	Branded Drug
13	19025093	RxNorm	Metoprolol Tartrate 50 MG Oral Tablet [Arbralene]	201333	Branded Drug
14	19025094	RxNorm	Metoprolol Tartrate 100 MG Oral Tablet [Betaloc]	201334	Branded Drug
15	19025095	RxNorm	Metoprolol Tartrate 100 MG Oral Tablet [Mepranix]	201335	Branded Drug
16	19025096	RxNorm	Metoprolol Tartrate 100 MG Oral Tablet [Arbralene]	201336	Branded Drug
17	19062037	RxNorm	Metoprolol 5 MG/ML Injectable Solution	250082	Clinical Drug
18	19083036	RxNorm	Metoprolol 100 MG	328470	Clinical Drug Comp
19	19091574	RxNorm	Metoprolol 5 MG/ML	335209	Clinical Drug Comp

Using the same code as in Administrative Example, use CROSS APPLY between the TVF that gets the drug ingredients and the TVF that gets all drugs for a given drug ingredient

Medication Summary

- Medication navigation and roll-up/aggregation is easy with the OMOP (UMLS) meta-data
- Relies on LocalDrugSID -> RxNorm mapping, and this is a work in progress
- Future discussions:
  - Drug Eras
  - Dose, Route, etc

VA OMOP - Documentation

Where do I go to find documentation, quality assessment reports, and general information about VA OMOP?

- www.vapulse.net VINCI OMOP Users Group

How can I request VA OMOP? (DART)

CDW DOMAIN CHECKLIST

Production

- Allergy
- Appointment
- Consult
- CPRS Orders
- Data Profiling

. . .

RAW*

- Bill Claims (CBO)
- CAPRI Audit Trail table
- CliniComp
- Echocardiogram
- Emergency Dept. Int. Software (EDIS)

. . .

Other Data

- PSSG Geocoded Files and ADUSH Enrollment Files
- OMOP Common Data Model v5 (CDW Production/Raw Source)

vinci@va.gov

Michael E. Matheny, MD, MS, MPH

Michael.matheny@va.gov

Michael.matheny@vanderbilt.edu

@MichaelEMatheny

```
USE [OMOP_V5]
GO

/* ------------------------- VINCI OMOP Iniative ---------------------------------
*/
/* Version 1.0 Michael Matheny */
/* How to use OMOP to build a basic computable phenotype for Heart Failure */
/* Educational Code Script to accompany CyberSeminar of 11/09/2017 by Michael
Matheny */
/*
Script Objectives:
        1. To generate a step-by-step process to computing a basic patient phenotype
within OMOP
        2. To highlight key important OMOP facts and conventions necessary to use
the CDM
        3. To underscore ease of use and small volume of code neceessary to execute
phenotype
        4. To promote community submission of valicated phenotypes for background
VINCI computation and
                persistence in CDM
        5. To promote use of community and VINCI developed table-valued functions
deployed SQL to abstract away
                a portion of CDM table and architectural in order to lower expertise
/ CDM knowledge required to use OMOP
*/

/*

OVERVIEW of OMOP Architecture:

At the highest level, OMOP rests on the foundation of the [CONCEPT] table, which is
a collection of all the UMLS controlled
vocabularies and additional cleaning and vocabularies contributed by the OHDSI
community, along with meta-data for those
concepts that defined relationships between the concepts (ancestors, etc).   The
concept library is ALWAYS a work in progress
and is updated and improved quarterly.  The related tables that help defined
[CONCEPT] relationships are [CONCEPT_ANCESTOR],
[CONCEPT_RELATIONSHIP],[CONCEPT_SYNONYM],[CONCEPT_CLASS]

As in CDW, there are OMOP tables that are Meta-Data tables, like the Dim tables, and
tables that are fact tables, like [RxOutPatFill]
or [SPatient] that contain clinical data linked to patients and clinical staff.

All OMOP Fact tables have an *_CONCEPT_ID field, and most have an
*_SOURCE_CONCEPT_ID field as well as a X_TYPE_CONCEPT_ID field.
These are key tables that link back to the [CONCEPT] table, and make the content of
that fact computable.

The [*_CONCEPT_ID] field is the primary content identifier in the target controlled
vocabulary for the type of clinical fact.
Examples:  SNOMED for ICD-9/ICD-10, RxNorm for Medications, NUCC for Provider
Specialty Type, etc.

The [*_SOURCE_CONCEPT_ID] field is the content identifier for the source controlled
vocabulary from the source record.  This maintains
fidelity with the conversion process from source to OMOP and allows users within a
specific instance of the CDM to retain the ability
to compute against the source vocabularies when they prefer to do so.
Examples in VA:  VUID/Va Product Code for RxOut.RxOutPatFill, ICD-9 for
Outpat.VDiagnosis, LOINC for ChemLab.PatientChemChem

The [*_TYPE_CONCEPT_ID] field identifies the general type of content of the fact.
```

This has slightly different flavors in
different domains:
Example 1: CONDITION_TYPE_CONCEPT_ID; this field dictates the inpatient vs
outpatient source of the administrative code and what position
(primary/secondary, 1, 2, 3, etc) the code was reported as.
Example 2: VISIT_TYPE_CONCEPT_ID: the visit is encoded from electronic health
record, or a claims record, etc.
*/

/*
PROCESS SUMMARY

In this example, we are using adminsitrative codes to build a phenotype, so being
able to identify those
patients with the appropriate codes and in specific temporal sequences is the task.
In order to do so, we must:
1. Identify and extract computable logic from a validated clinical phenotype
2. translate our administrative code definition into OMOP CONCEPTS
3. Identify the correct OMOP Fact tables to use
4. Handle any specific logic regarding combinations or counts of administrative
codes necessary to make the phenotype definition
*/

/*
VINCI OMOP Meta-Data Management Script Library

In order to help users quickly and accurately use OMOP in their work, we have
translated a large library previously developed
for OMOP Version 4 by the OHDSI user community into OMOP V5 compatible scripts and
installed them in VINCI for re-use. In addition,
we will develop additional scripts as needed by the VINCI community to support quick
re-use of OMOP meta-data. This allows
users that do not want to or are unable to (not currently installable in VA
environment) use the ATLAS and OHDSI user interface
tools to be able to easily use similar functions directly in SQL.

DOCUMENTATION: Please see VA Pulse - VINCI OMOP site for all documentation regarding
the script library.
DIRECT ACCESS: You can access these scripts from the your relevant VINCI RB server
in CDWWORK, Programmability, Table-valued Functions

this is an example of how the table-valued meta-data support procesures are written,
whith common parameters:
```
SELECT * FROM [OMOPV5].[tvf_C05_TranslateSourceCodeToConditionConcepts] (
   <@SOURCE_CODE, varchar(20),>
  ,<@SOURCE_VOCABULARY_ID, varchar(20),>
  ,<@INDEX_DATE, date,>)
```

SOURCE_CODE is the source value int he vocabulary, example for ICD-9-CM, '250.0'
would be a source code
SOURCE_VOCABULARY_ID:

First, look in the [OMOPV5].[VOCABULARY] table to determine what the name of the
vocabulary is that you want to translate

Examples potentially useful in this script:

```
---------- ADMIN CODES ----------
VOCABULARY_ID   VOCABULARY_NAME
Source:
CPT4                    Current Procedural Terminology version 4 (AMA)
HCPCS                   Healthcare Common Procedure Coding System (CMS)
ICD10                   International Classification of Diseases, Tenth Revision
```

(WHO)
ICD10CM International Classification of Diseases, Tenth Revision, Clinical Modification (NCHS)
ICD10PCS ICD-10 Procedure Coding System (CMS)
ICD9CM International Classification of Diseases, Ninth Revision, Clinical Modification, Volume 1 and 2 (NCHS)
ICD9Proc International Classification of Diseases, Ninth Revision, Clinical Modification, Volume 3 (NCHS)
Target:
SNOMED Systematic Nomenclature of Medicine - Clinical Terms (IHTSDO)

Note, it is helpful to know what the 'standard' vocabulary is for OMOP in a particular domain. For administrative codes
that target vocabulary is SNOMED-CT, regardless of the source vocabulary mapping. This is important for use across health
systems, because different source systems use different codes, and a consensus mapping to SNOMED-CT allows code to run consistently
across multiple CDM instances. In theory, this also allows a direct cross-walk between ICD-9 era codes and ICD-10 era codes, but
in practice, these crosswalks are incomplete because they are still being developed and matured.

INDEX_DATE: this can be null if you want ALL codes, which is our recommended default convention. If you set a date
this is the date which you are enforcing all codes to be active as defined by the vocabulary owner. Please note that in the
VA, inactive codes are still used because of systematic differences in the healthcare system from the private system. Including inactive
codes will not create any errors in downstream SQL execution if they are not used.
*/

/*
The following example is how to use a TVF script to retreive the OMOP concept ID for a single ICD-9 Code, along with
the translation of that code to SNOMED CONCEPT ID(s) or other target vocabulary (works for all domains):
SELECT * FROM [OMOPV5].[tvf_V01_TranslateSourceCodeToConditionConcepts]
('250.01','ICD9CM',null)

The following example is how to use the TVF script to get all OMOP CONCEPT_ID matches for a single ICD-9 code, this is
the core useful TVF that helps users build CONCEPT_ID filter lists for their needs:
SELECT * FROM [OMOPV5].[tvf_V02_SoureCodeMappingList] ('250.11','ICD9CM',null)

IMPORTANT CONVENTION: This TVF script returns all the CONCEPT_ID's from the source AND target vocabularies for the code, so that the
resulting list can be applied to the relevant *_CONCEPT_ID field OR the
*_SOURCE_CONCEPT_ID field and should return identical results
in the VA OMOP CDM.

In almost all cases, users need to build lists of codes from a small to large list of source vocabulary codes, not a single
code. Microsoft SQL has an easy programming convention to do that for TVF function scripts.

The following is a toy example of 3 diabetes codes, and an easy convention for how to build user-defined code filter lists
for use with the TVF functions. Please note these can be persisteted in your working database for ease of re-use. In that
case you would create a table such as [Temp].[CodeList] instead of #TempList.
*/

```
CREATE table #TempList (
        [Source_Code] [varchar](50) null
        ,[Vocabulary_ID] [varchar](50) null
        ,[DateFilter] [date] null
        )

INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('250.01','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('250.11','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('250.21','ICD9CM',null);

/*
The 'trick' to using TVF with lists of codes is to use the CROSS APPLY SQL function
as noted below:

select *
from #TempList [X]
        cross apply [OMOPV5].[tvf_V02_SoureCodeMappingList]
([X].[Source_Code],[X].[Vocabulary_ID],[X].[DateFilter]);

You can un-comment this section or highlight and run this code directly to see the
results.
*/

/* this clears the code list table to prepare for a new example */
truncate table #TempList

/*
DEVELOPMENT OF a Heart Failure Phenotype from Administrative Data

  Based on Go, et al. Circulation 2006;113(23):2713-2723 that used combo
inpatient/outpatient codes
                with >= 1 hospitalization with CHF as primary code
                or >= 2 any codes (inpatient/ED/outpatient primary secondary)
                PPV 97% compared to physician review
                Original Publication Codes: 398.91, 402.01, 402.11, 402.91, 428.0,
428.1, or 428.9

                Because of the age of the publication, clinical review of possible
ICD-9 heart failure codes published in
                other manuscripts (Mini-Sentinel HF Report) and assessment of volume
of codes per year used in
                VA suggested addition of some closely related codes.

                Please note that adding codes de-validates the phenotype from the
original anchor publication, but is done here
                to create the example as used in a prior research project.
                Community goverance and careful documentation transparence are
required for community re-use of computable phenotypes.
*/

/* Load CHF Codes */
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.0','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.1','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
```

```
VALUES('428.2','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.20','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.21','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.22','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.23','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.3','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.30','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.31','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.32','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.33','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.4','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.40','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.41','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.42','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.43','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('428.9','ICD9CM',null);

INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('398.91','ICD9CM',null);

INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('402.01','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('402.11','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('402.91','ICD9CM',null);

INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('404.01','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('404.03','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('404.11','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('404.13','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('404.91','ICD9CM',null);
INSERT INTO #TempList ([Source_Code],[Vocabulary_ID],[DateFilter])
VALUES('404.93','ICD9CM',null);

/* this creates the CONCEPT_ID filter that will be used to find all the heart
failure administrative
codes in the OMOP clinical fact table(s). */

select *
INTO #TempFilter
from #TempList [X]
        cross apply [OMOPV5].[tvf_V02_SoureCodeMappingList]
```

```
([X].[Source_Code],[X].[Vocabulary_ID],[X].[DateFilter]);
create clustered index [ind_1223] on #TempFilter ([CONCEPT_ID]);

/*
It is not required to stage the coding with interim temp tables, but after
evaluation of the code with and without
creating a new temp table, it was found that the exeuction speed of creating a temp
table from the filter list and
 indexing it was substantially faster than directly including this code segment in
the fact filter join.
*/


drop table #TempCondition

SELECT TOP 100000 [A].[PERSON_ID]
      ,[CONDITION_CONCEPT_ID]
          ,[CONDITION_TYPE_CONCEPT_ID]
          /* these omop concepts are all the primary/1st position condtion type
concepts
          that are either for sure inpatient or ambiguous and may be inpatient */
          /*  select * from omop_v5.omopv5.concept where vocabulary_id = 'condition
type' */
          ,CASE WHEN [CONDITION_TYPE_CONCEPT_ID] in
(38000183,38000184,38000199,38000200,44786627,44786628,45756835,45756843) THEN 1
                ELSE 0 END as [PrimaryCode]
                /* these omop concepts are the inaptient and inpatient obs for visit
codes */
                 /*  select * from omop_v5.omopv5.concept where vocabulary_id =
'visit' */
          ,CASE WHEN [Visit].[VISIT_CONCEPT_ID] in (9201,800000001) THEN 1 ELSE 0
END as [InpatientCode]
          /* OMOP has both a date field and date/time field, in this case we want
only date level resolution */
      ,[CONDITION_START_DATE]
          ,[A].[VISIT_OCCURRENCE_ID]
INTO #TempCondition
FROM [OMOP_V5].[OMOPV5].[CONDITION_OCCURRENCE] a
INNER JOIN #TempFilter b
on a.[CONDITION_CONCEPT_ID] = b.concept_id
left join [OMOP_V5].[OMOPV5].[VISIT_OCCURRENCE] [Visit]
on [A].[VISIT_OCCURRENCE_ID] = [Visit].[VISIT_OCCURRENCE_ID]
create clustered index [ind_1223] on #TempCondition ([VISIT_OCCURRENCE_ID]);

/* Collapse the data into person date records stratified by primary inpatient (only
requires 1 code) or all others (requires 2 codes) */
/* this finalizes the prep for the logic of a single inpaient primary or two other
codes
        this also removes multiple codes from the same day (count only one code per
day of each type ) */
select
        [PERSON_ID]
        ,[InpatientPrimary]
        ,[CONDITION_START_DATE]
INTO #TempDistinctPersonDates
FROM (
        select
                [PERSON_ID]
                ,CASE WHEN [PrimaryCode] = 1 and [InpatientCode] = 1 THEN 1 ELSE 0
END as [InpatientPrimary]
                ,[CONDITION_START_DATE]
        FROM #TempCondition
        ) x
```

```
group by        [PERSON_ID]
        ,[InpatientPrimary]
        ,[CONDITION_START_DATE]

select top 100 * from #TempDistinctPersonDates
order by person_id, CONDITION_START_DATE asc

/* find the minimum date for each person_id and secondary code, so you can exclude
it */
SELECT
        [PERSON_ID]
        ,min([CONDITION_START_DATE]) as [MIN_CONDITION_START_DATE]
INTO #MinDateTwoCodesRequired
FROM #TempDistinctPersonDates
where [InpatientPrimary] = 0
group by [PERSON_ID]

SELECT
        [PERSON_ID]
        ,min([QUALIFYING_START_DATE]) as [QUALIFYING_START_DATE]
INTO #FinalPhenotypeDate
FROM (
        SELECT
                [PERSON_ID]
                ,min([CONDITION_START_DATE]) as [QUALIFYING_START_DATE]
        FROM #TempDistinctPersonDates
        where [InpatientPrimary] = 1
        GROUP BY [PERSON_ID]
        UNION ALL
        SELECT
                [PERSON_ID]
                ,min([CONDITION_START_DATE]) as [QUALIFYING_START_DATE]
        FROM (
                SELECT
                        a.[PERSON_ID]
                        ,a.[CONDITION_START_DATE]
                FROM #TempDistinctPersonDates a
                left join #MinDateTwoCodesRequired b
                on a.PERSON_ID = b.PERSON_ID
                and a.CONDITION_START_DATE = b.MIN_CONDITION_START_DATE
                where a.InpatientPrimary = 0
                 and b.PERSON_ID is null
                ) x
        group by [PERSON_ID]
        ) final
group by [PERSON_ID]

/* check that the logic works as intended */
SELECT a.*,b.[QUALIFYING_START_DATE]
from #TempDistinctPersonDates  a
inner join  #FinalPhenotypeDate b
on a.person_id = b.person_id
order by person_id, CONDITION_START_DATE
/* upon inspection, the logic worked, the phenotype qualifying date was the 2nd date
in the list
except when the 1st date was an Inpatient Primary code */


/* Lastly, we want to represent this data in an OMOP compliant way, so that it could
be persisted for all users of OMOP,
and to promote transparency for logic, and to allow use of all OHDSI tools that
leverage the COHORT table.
```

```
COHORT Table specification:
COHORT_DEFINITION_ID int - this is an auto-integer field, that gets added as new
cohorts / phenotypes get added
SUBJECT_ID - this is the key to the linking OMOP fact table, right now this can be
PERSON, PROVIDER, VISIT_OCCCURRENCE (but only one per definition)
COHORT_START_DATE - date the SUBJECT_ID enters the cohort/phenotype o interest
COHORT_END_DATE - date the SUBJECT_ID leaves the cohort/phenotype of interest.  For
some chronic phenotypes, a subject never
        leaves, but for the purposes of censoring, this is supposed to not be null,
and would be the end of the data collection window

COHORT_DEFINITION Table Specification:
COHORT_DEFINITION_ID - links to the primary key in COHORT table
COHORT_DEFINITION_NAME - text name field for cohort/phenotype
COHORT_DEFINITION_DESCRIPTION - description, references, validation, logic summary,
etc
DEFINITION_TYPE_CONCEPT_ID - what kind of cohort definition the record represents
COHORT_DEFINIITION_SYNTAX - logic / programmable code to operationalize the cohort
definition
SUBJECT_CONCEPT_ID - This defines the domain for the SUBJECT_ID (PERSON, PROVIDER,
VISIT_OCCURRENCE)
COHORT_INITIATION_DATE - start date of cohort build

*/

/* Please NOTE!  For your own projects, you will have to build a custom COHORT and
COHORT_DEFINITION table, build a view
that unions the stock VINCI tables and your custom table in your working database,
and then populate additional cohorts
beyond the VINCI pre-computed ones */
INSERT INTO [Temp].[COHORT] (
        [COHORT_DEFINITION_ID]
        ,[SUBJECT_ID]
        ,[COHORT_START_DATE]
        ,[COHORT_END_DATE]
        )
SELECT
        2017110701 as [COHORT_DEFINITION_ID] /* made up integer, must just make sure
cohort uses the number and it doesn't collide with other assignments */
        ,[PERSON_ID] as [SUBJECT_ID]
        ,[QUALIFYING_START_DATE] as [COHORT_START_DATE]
        ,'11/07/2017' as [COHORT_END_DATE]
FROM #FinalPhenotypeDate

INSERT INTO [Temp].[COHORT_DEFINITION] (
        [COHORT_DEFINITION_ID]
        ,[COHORT_DEFINITION_NAME]
        ,[COHORT_DEFINITION_DESCRIPTION]
        ,[DEFINITION_TYPE_CONCEPT_ID]
        ,[COHORT_DEFINITION_SYNTAX]
        ,[SUBJECT_CONCEPT_ID]
        ,[COHORT_INITIATION_DATE]
        )
VALUES (
        2017110701  /* [COHORT_DEFINITION_ID] */
        ,'Heart Failure (Modified Go 2006)' /* [COHORT_DEFINITION_NAME] */
        ,'   Based on Go, et al. Circulation 2006;113(23):2713-2723 that used combo
inpatient/outpatient codes
                with >= 1 hospitalization with CHF as primary code
                or >= 2 any codes (inpatient/ED/outpatient primary secondary)
                PPV 97% compared to physician review
                Original Publication Codes: 398.91, 402.01, 402.11, 402.91, 428.0,
428.1, or 428.9
```

```
                 Presentation+-+CyberSeminar+Code+2017-11-09++extra
                Implemented Codes:  See COHORT_DEFINITION_SYNTAX, using additional
codes from Mini-Sentinel HF Review
                ' /*  [COHORT_DEFINITION_DESCRIPTION] */
                ,-1 /* DEFINITION_TYPE_CONCEPT_ID */  /* future work */
                ,'<INSERT PROGRAMMING CODE>' /* [COHORT_DEFINITION_SYNTAX] */
                ,56 /* [SUBJECT_CONCEPT_ID] */ /* this is the meta-data concept for
PERSON, defines the domain of SUBJECT_ID in COHORT */
                ,'11/07/2017' /* [COHORT_INITIATION_DATE] */
        )
```