OHDSI Collaborator Meeting: **Unit & Regression Testing of your Common Data Model**

20-FEB-2018

Erica Voss

Clair Blacketer / Ajit Londhe / Jamie Weaver

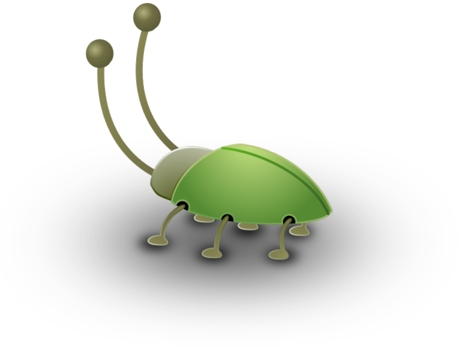
Today's Discussion

- High-level Testing Terminology

- Life Cycle of Testing a CDM
  - White Rabbit / Rabbit In a Hat
  - Testing Framework
  - How to Execute Testing Process

- Janssen Specific Examples

Testing Terms



- **Unit Testing**
  - – individual aspects of your ETL requirements are tested

- **Regression Testing**
  - – Ensuring that previously developed and tested aspects of a ETL continue to work
  - – Building up a series of unit tests allow you to regression test

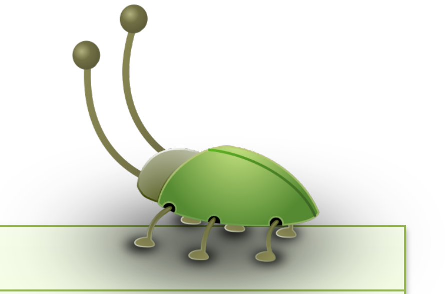
Unit Testing Example

- **ETL States:**
  Person born in the future should be excluded from the CDM

- **Unit Test:**
  Person 1234 with date of birth year 2099

- **CDM Builder Behavior:**
  Expect no Person 1234 in the CDM

Regression Testing Examples



Person with two genders is excluded.

Person with two birth years >2 yrs apart is excluded.

Person with two birth years <2 yrs apart is kept with last birth year selected.

Person born before 1900 is excluded.

Person born in 2099 is excluded.

Person born in 2014 but enrolled in 2012 is excluded.

Person born in 2013 but enrolled in 2012 is kept, latest birth year taken.

Person with two enrollment_detail records has one person record.

Person with sex=3 is excluded.

Person has record with sex=3 but last record has sex=1, person is kept.

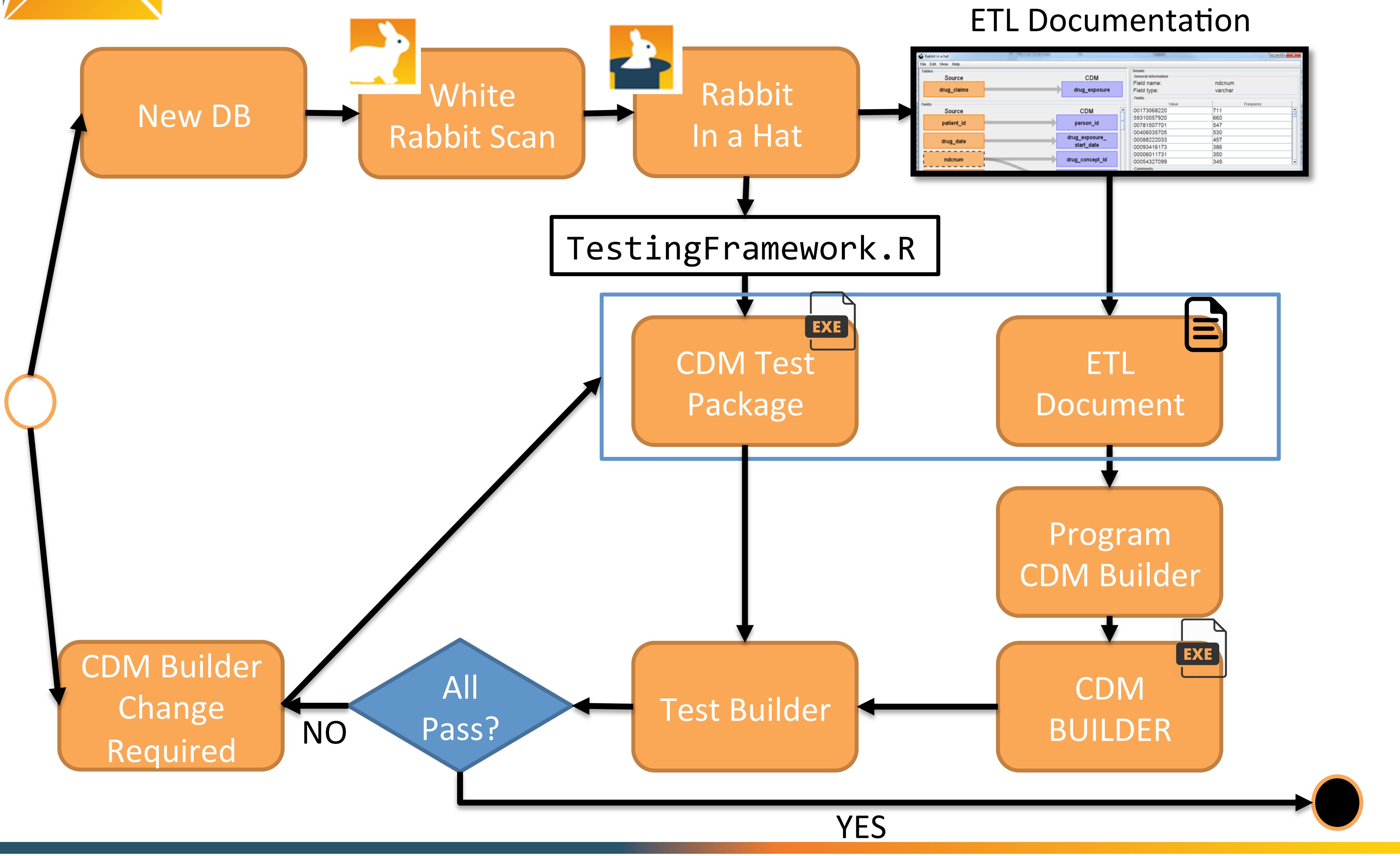
Person born the same year as enrolled, use first enrollment month to impute month of birth and day of birth.

Person with Rx benefits is kept.

Person without Rx benefits is excluded.

Person with last enrollment_detail record that has egeoloc=11 gets associated to NJ.

Life Cycle of Testing a CDM



Rabbit in a hat

File Edit View Help

Tables

Source	CDM
drug_claims	drug_exposure

Fields

Source	CDM
patient_id	person_id
drug_date	drug_exposure_ start_date
ndcnum	drug_concept_id

Details

General information

Field name: ndcnum

Field type: varchar

Fields

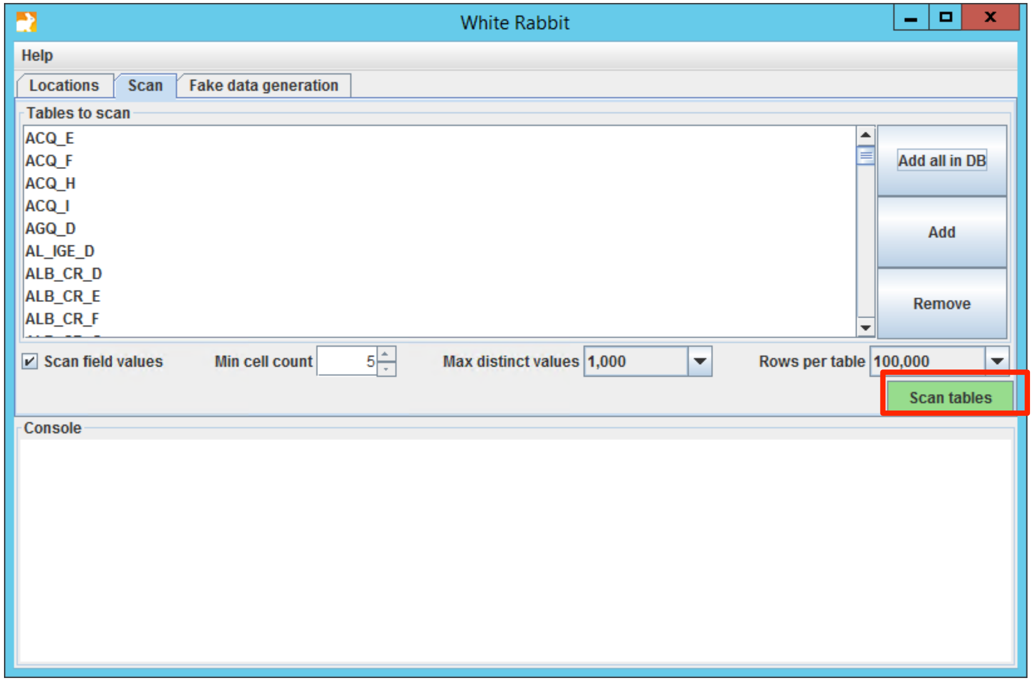
Value	Frequency
00173068220	711
59310057920	660
00781507701	547
00406035705	530
00088222033	457
00093416173	386
00006011731	350
00054327099	349

Comments

White Rabbit Scan

- Scans source data providing detailed information on the tables, fields, and values that appear in a field

- Example form NHANES:



Tables to scan

ACQ_E
ACQ_F
ACQ_H
ACQ_I
AGQ_D
AL_IGE_D
ALB_CR_D
ALB_CR_E
ALB_CR_F

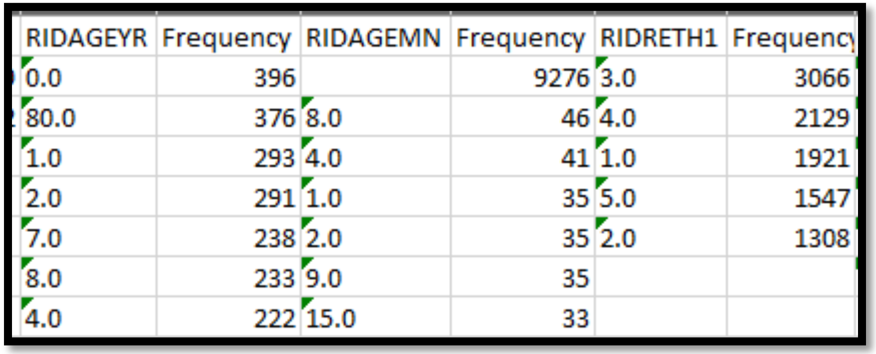
☑ Scan field values Min cell count 5 Max distinct values 1,000 Rows per table 100,000

After connecting on the location tab the scan tab generates the scan report

White Rabbit Scan

- Example form NHANES:

DEMO_I



RIDAGEYR	Frequency	RIDAGEMN	Frequency	RIDRETH1	Frequency
0.0	396		9276	3.0	3066
80.0	376	8.0	46	4.0	2129
1.0	293	4.0	41	1.0	1921
2.0	291	1.0	35	5.0	1547
7.0	238	2.0	35	2.0	1308
8.0	233	9.0	35		
4.0	222	15.0	33		

RIDAGERY = age in years

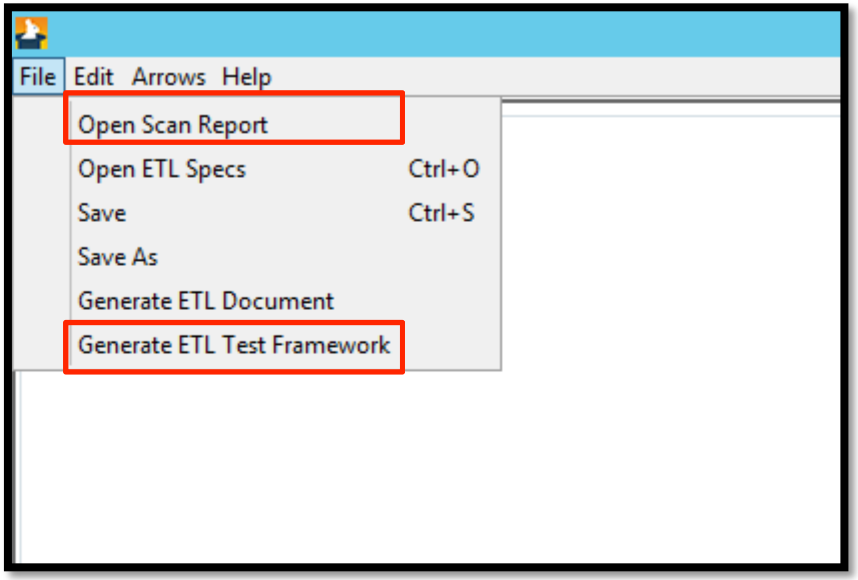
RIDAGEMN = Age in Month
(for <25 months of age)

RIDRETH1 = race-ethnicity

"Demographic Variables and Sample Weights (DEMO_I)." Retrieved 2018.02.14, from https://wwwn.cdc.gov/Nchs/Nhanes/2015-2016/DEMO_I.htm.

Rabbit in the Hat

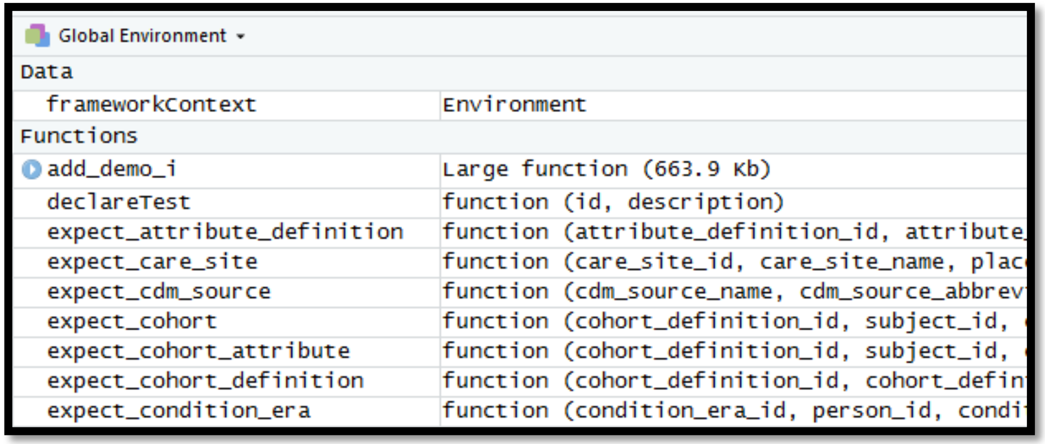
- Using the scan report you can automatically generate functions to help you build test cases.



Open the scan report created in WhiteRabbit and then click 'Generate ETL Test Framework'

Rabbit in the Hat

- Source the TestingFramework.R in RStudio to make the test functions available



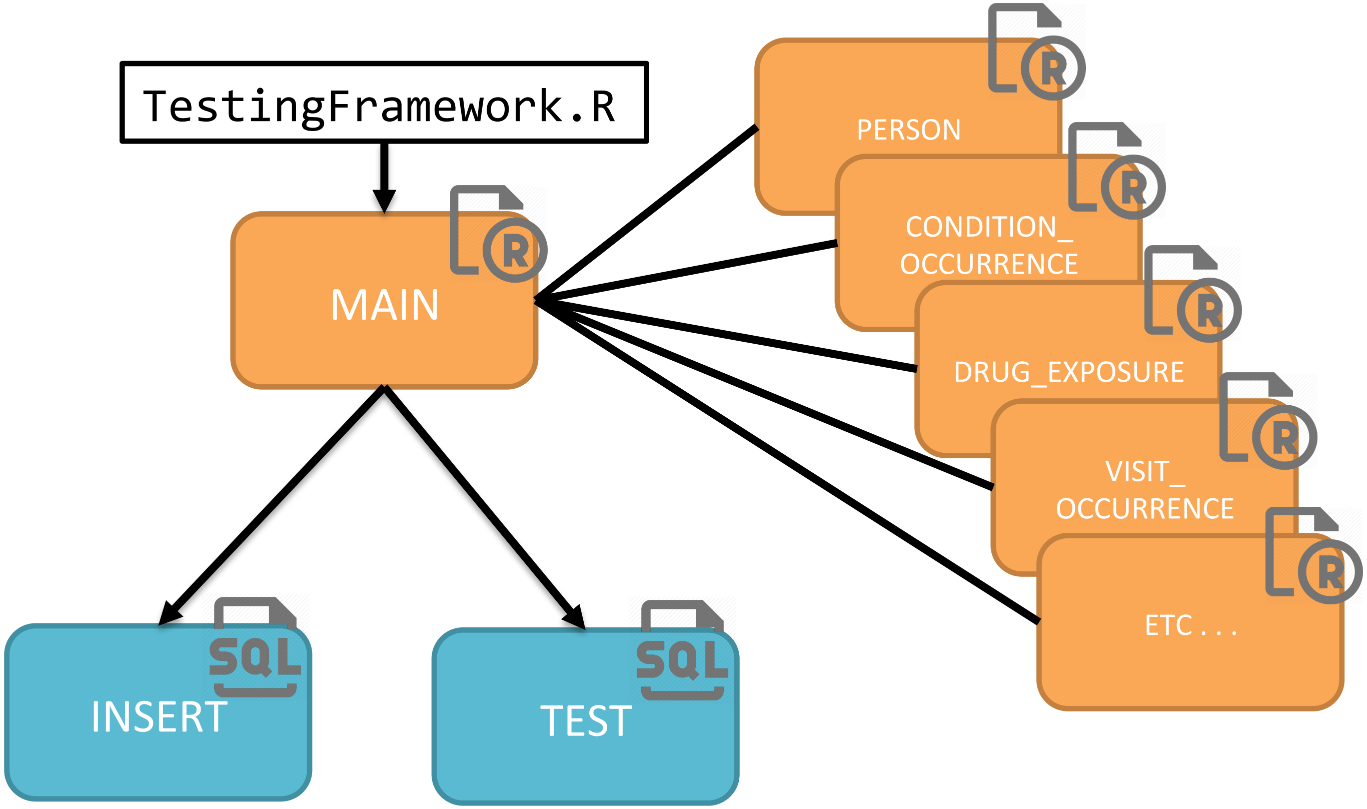
```
Global Environment ▾
Data
  frameworkContext            Environment
Functions
  add_demo_i                  Large function (663.9 Kb)
  declareTest                 function (id, description)
  expect_attribute_definition function (attribute_definition_id, attribute_
  expect_care_site            function (care_site_id, care_site_name, plac
  expect_cdm_source           function (cdm_source_name, cdm_source_abbrev
  expect_cohort               function (cohort_definition_id, subject_id,
  expect_cohort_attribute     function (cohort_definition_id, subject_id,
  expect_cohort_definition    function (cohort_definition_id, cohort_defin
  expect_condition_era        function (condition_era_id, person_id, condi
```

ADD_ functions allow you to add data to raw tables to generate your test cases

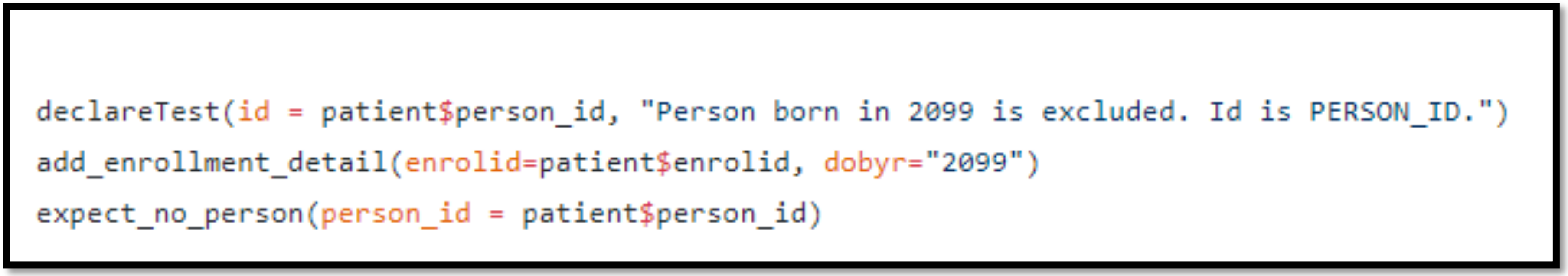
EXPECT_ functions allow you to define what you expect in the CDM

CDM Test Package

- R package that stitches all your tests together using functions from TestingFramework.R



CDM Test Package:
Example Test Case 1



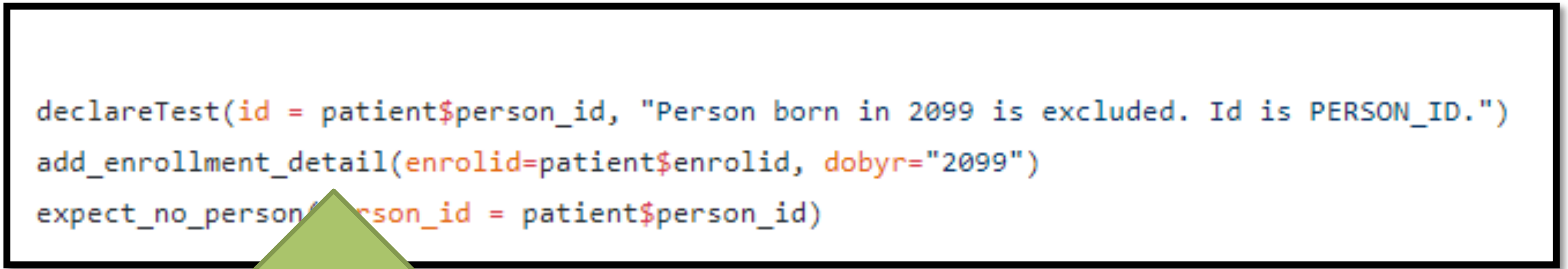
```
declareTest(id = patient$person_id, "Person born in 2099 is excluded. Id is PERSON_ID.")
add_enrollment_detail(enrolid=patient$enrolid, dobyr="2099")
expect_no_person(person_id = patient$person_id)
```

- Example, ETL describes that if a person is born in the future they should be excluded from the CDM

- ADD_ data to raw data tables to mimic this behavior, this adds to INSERT.SQL

- EXPECT_ to state what you expect to occur in CDM, this adds to TEST.SQL

CDM Test Package: Example Test Case 1

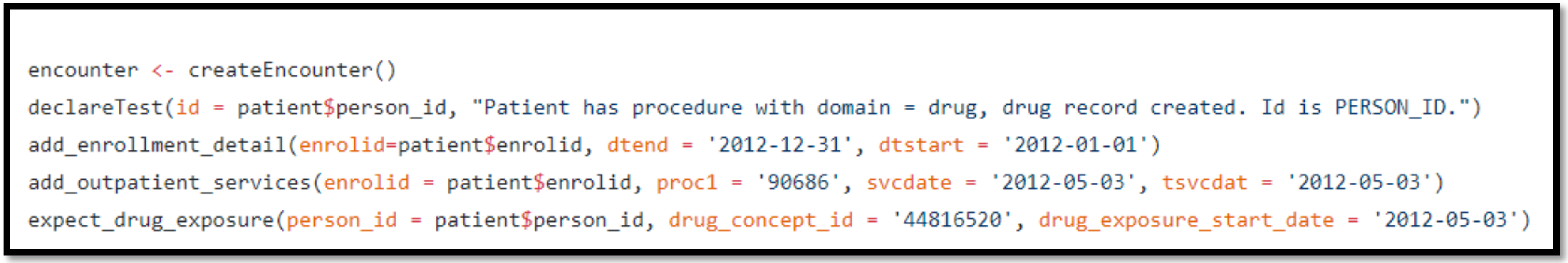


```
declareTest(id = patient$person_id, "Person born in 2099 is excluded. Id is PERSON_ID.")
add_enrollment_detail(enrolid=patient$enrolid, dobyr="2099")
expect_no_person(  rson_id = patient$person_id)
```

Rabbit in a Hat Genius!
Where are all the other values the table needs? The scan report just populates them with the most common value.

- Exampl describes that if a person is born in the excluded from the CDM
- tables to mimic this behavior, L
- EXPECT_ to state what you expect to occur in CDM, this adds to TEST.SQL

CDM Test Package: Example Test Case 2



```
encounter <- createEncounter()
declareTest(id = patient$person_id, "Patient has procedure with domain = drug, drug record created. Id is PERSON_ID.")
add_enrollment_detail(enrolid=patient$enrolid, dtend = '2012-12-31', dtstart = '2012-01-01')
add_outpatient_services(enrolid = patient$enrolid, proc1 = '90686', svcdate = '2012-05-03', tsvcdat = '2012-05-03')
expect_drug_exposure(person_id = patient$person_id, drug_concept_id = '44816520', drug_exposure_start_date = '2012-05-03')
```

- Example, ETL describes that a CPT 90686-"influenza virus vaccine" that the OMOP Vocabulary associates with a concept in the drug domain

- ADD_ data to raw data tables to mimic this behavior, this adds to INSERT.SQL

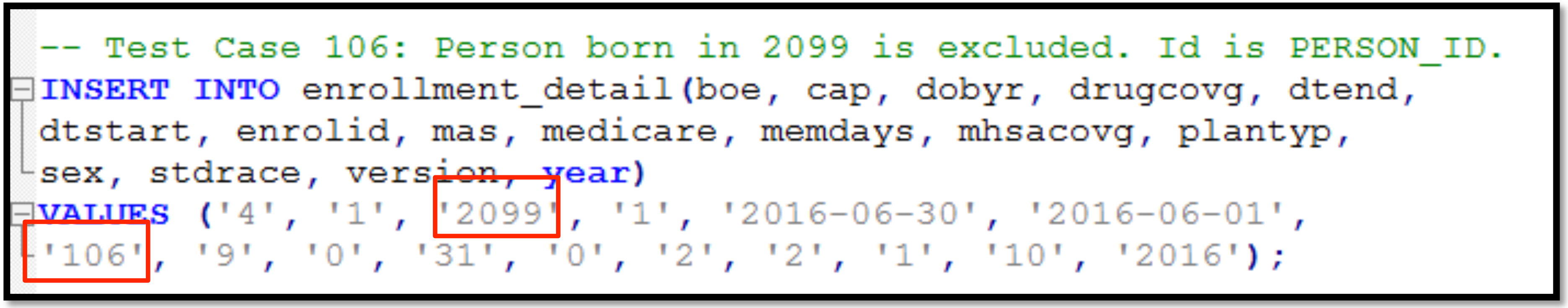
- EXPECT_ to state what you expect to occur in CDM, this adds to TEST.SQL

Execute Testing

1. INSERT.SQL populates raw DB
2. Run CDM Builder
3. TEST.SQL tests new CDM
4. Review test results
5. Augment test cases or CDM Builder till all tests pass

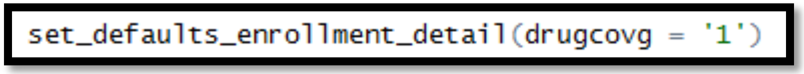
Execute Testing:
1) R populates raw DB

SQL INSERT



```
-- Test Case 106: Person born in 2099 is excluded. Id is PERSON_ID.
INSERT INTO enrollment_detail(boe, cap, dobyr, drugcovg, dtend,
dtstart, enrolid, mas, medicare, memdays, mhsacovg, plantyp,
sex, stdrace, version, year)
VALUES ('4', '1', '2099', '1', '2016-06-30', '2016-06-01',
'106', '9', '0', '31', '0', '2', '2', '1', '10', '2016');
```

- Test Case 1, we wanted to test a birth year of 2099
- The White Rabbit and the TestingFramework have filled in the rest of the values with the most common value from the scan
  - Sometimes you may additionally need to set multiple values in the test case (i.e. set the dates so things align correctly)
  - You can override the most common value with defaults in the TestingFramework

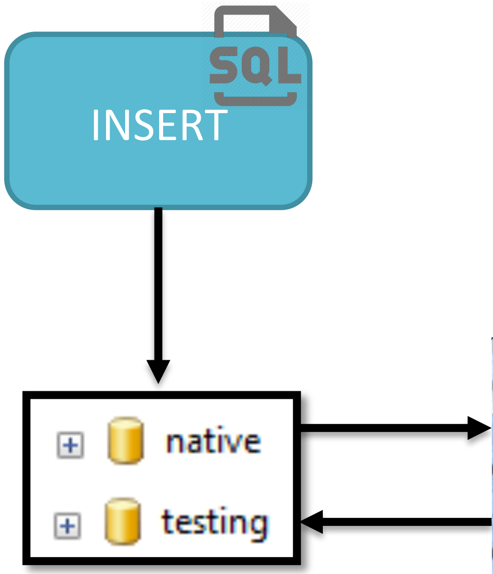


```
set_defaults_enrollment_detail(drugcovg = '1')
```

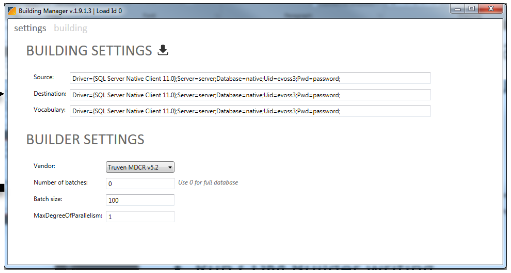
Execute Test Cases:
2) Run CDM Builder

- **NATIVE** (a database with all the raw tables has been set up and populated with your test cases in previous step)

- Run CDM Builder writing to CDM tables in **TESTING** (a database with a blank CDM schema for the builder to insert the resulting data)



CDM_BUILDER



BUILDING SETTINGS

Source: Driver={SQL Server Native Client 11.0};Server=server;Database=native;Uid=evoss3;Pwd=password;

Destination: Driver={SQL Server Native Client 11.0};Server=server;Database=native;Uid=evoss3;Pwd=password;

Vocabulary: Driver={SQL Server Native Client 11.0};Server=server;Database=native;Uid=evoss3;Pwd=password;

BUILDER SETTINGS

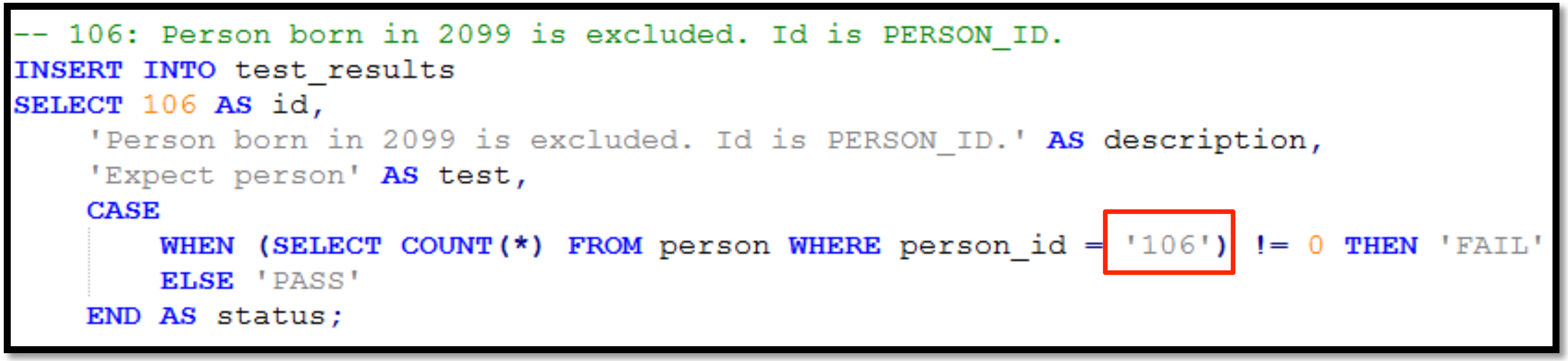
Vendor: Truven MDCR v5.2

Number of batches: 0 *Use 0 for full database*

Batch size: 100

MaxDegreeOfParallelism: 1

Execute Testing: 3) R tests new CDM

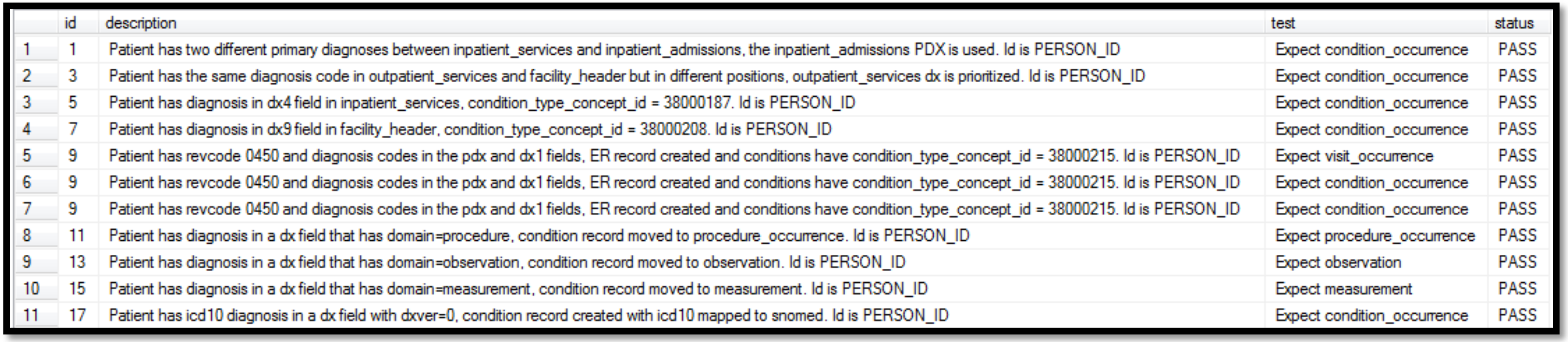


```
-- 106: Person born in 2099 is excluded. Id is PERSON_ID.
INSERT INTO test_results
SELECT 106 AS id,
    'Person born in 2099 is excluded. Id is PERSON_ID.' AS description,
    'Expect person' AS test,
    CASE
        WHEN (SELECT COUNT(*) FROM person WHERE person_id = '106') != 0 THEN 'FAIL'
        ELSE 'PASS'
    END AS status;
```

- The EXPECT_ functions have written tests similar to the above

- If we find PERSON_ID 109 in the CDM who was born in 2099 the test will fail

Execute Testing:
4) Review test results

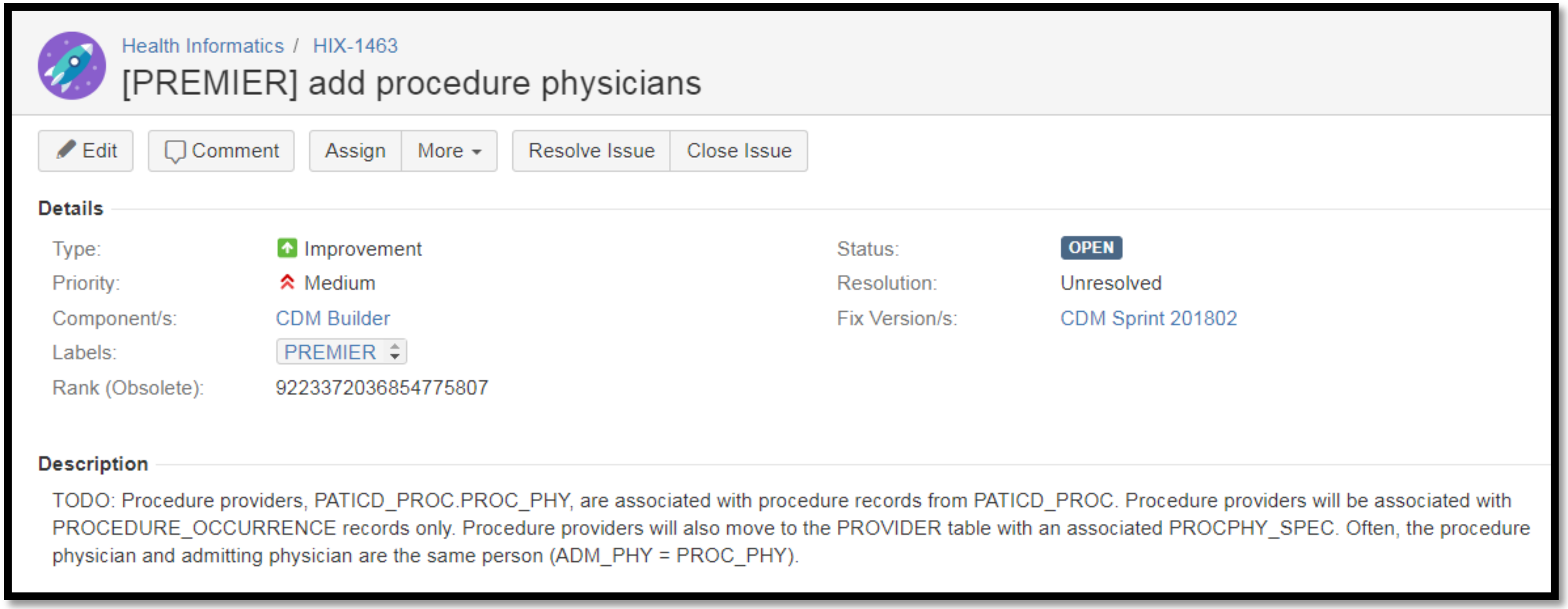


	id	description	test	status
1	1	Patient has two different primary diagnoses between inpatient_services and inpatient_admissions, the inpatient_admissions PDX is used. Id is PERSON_ID	Expect condition_occurrence	PASS
2	3	Patient has the same diagnosis code in outpatient_services and facility_header but in different positions, outpatient_services dx is prioritized. Id is PERSON_ID	Expect condition_occurrence	PASS
3	5	Patient has diagnosis in dx4 field in inpatient_services, condition_type_concept_id = 38000187. Id is PERSON_ID	Expect condition_occurrence	PASS
4	7	Patient has diagnosis in dx9 field in facility_header, condition_type_concept_id = 38000208. Id is PERSON_ID	Expect condition_occurrence	PASS
5	9	Patient has revcode 0450 and diagnosis codes in the pdx and dx1 fields, ER record created and conditions have condition_type_concept_id = 38000215. Id is PERSON_ID	Expect visit_occurrence	PASS
6	9	Patient has revcode 0450 and diagnosis codes in the pdx and dx1 fields, ER record created and conditions have condition_type_concept_id = 38000215. Id is PERSON_ID	Expect condition_occurrence	PASS
7	9	Patient has revcode 0450 and diagnosis codes in the pdx and dx1 fields, ER record created and conditions have condition_type_concept_id = 38000215. Id is PERSON_ID	Expect condition_occurrence	PASS
8	11	Patient has diagnosis in a dx field that has domain=procedure, condition record moved to procedure_occurrence. Id is PERSON_ID	Expect procedure_occurrence	PASS
9	13	Patient has diagnosis in a dx field that has domain=observation, condition record moved to observation. Id is PERSON_ID	Expect observation	PASS
10	15	Patient has diagnosis in a dx field that has domain=measurement, condition record moved to measurement. Id is PERSON_ID	Expect measurement	PASS
11	17	Patient has icd10 diagnosis in a dx field with dxver=0, condition record created with icd10 mapped to snomed. Id is PERSON_ID	Expect condition_occurrence	PASS

- All TEST.SQL to write out to a TEST_RESULTS table in your CDM, review the failures

- Failures may indicate either a bug or a poorly written test case

Execute Test Cases:
5) Augment Test Cases/Builder



[PREMIER] add procedure physicians

Details

Type:	Improvement	Status:	OPEN
Priority:	Medium	Resolution:	Unresolved
Component/s:	CDM Builder	Fix Version/s:	CDM Sprint 201802
Labels:	PREMIER		
Rank (Obsolete):	9223372036854775807		

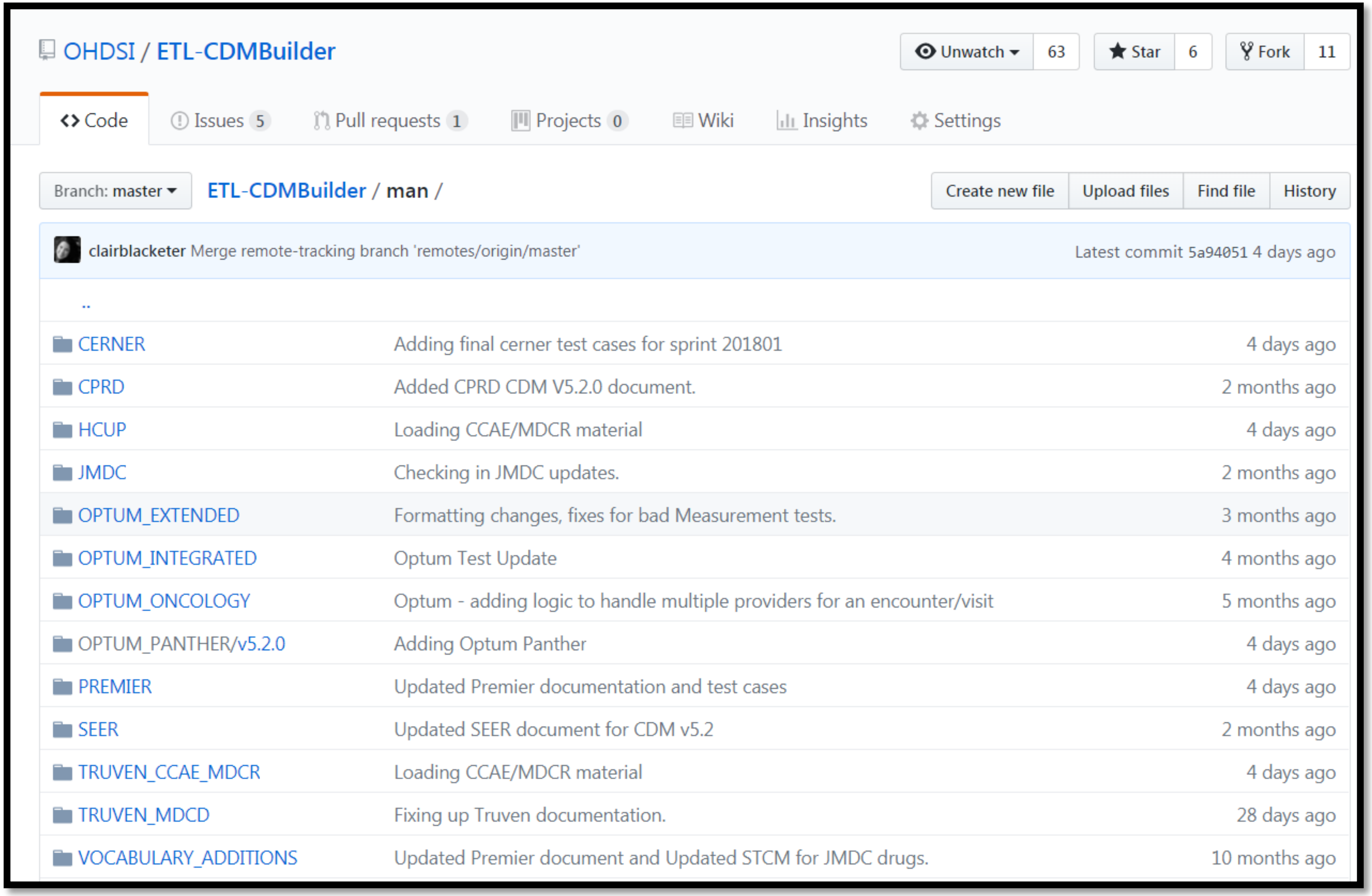
Description

TODO: Procedure providers, PATICD_PROC.PROC_PHY, are associated with procedure records from PATICD_PROC. Procedure providers will be associated with PROCEDURE_OCCURRENCE records only. Procedure providers will also move to the PROVIDER table with an associated PROCPHY_SPEC. Often, the procedure physician and admitting physician are the same person (ADM_PHY = PROC_PHY).

- If changes are required to your Builder having some sort of issue tracking will help keep you organized and also help track what changes are within each CDM Builder release

Some Janssen R&D Specific Ideas

Janssen ETLs and Testing are Open Source



OHDSI / ETL-CDMBuilder

ETL-CDMBuilder / man /

clairblacketer Merge remote-tracking branch 'remotes/origin/master' Latest commit 5a94051 4 days ago

..		
CERNER	Adding final cerner test cases for sprint 201801	4 days ago
CPRD	Added CPRD CDM V5.2.0 document.	2 months ago
HCUP	Loading CCAE/MDCR material	4 days ago
JMDC	Checking in JMDC updates.	2 months ago
OPTUM_EXTENDED	Formatting changes, fixes for bad Measurement tests.	3 months ago
OPTUM_INTEGRATED	Optum Test Update	4 months ago
OPTUM_ONCOLOGY	Optum - adding logic to handle multiple providers for an encounter/visit	5 months ago
OPTUM_PANTHER/v5.2.0	Adding Optum Panther	4 days ago
PREMIER	Updated Premier documentation and test cases	4 days ago
SEER	Updated SEER document for CDM v5.2	2 months ago
TRUVEN_CCAE_MDCR	Loading CCAE/MDCR material	4 days ago
TRUVEN_MDCD	Fixing up Truven documentation.	28 days ago
VOCABULARY_ADDITIONS	Updated Premier document and Updated STCM for JMDC drugs.	10 months ago

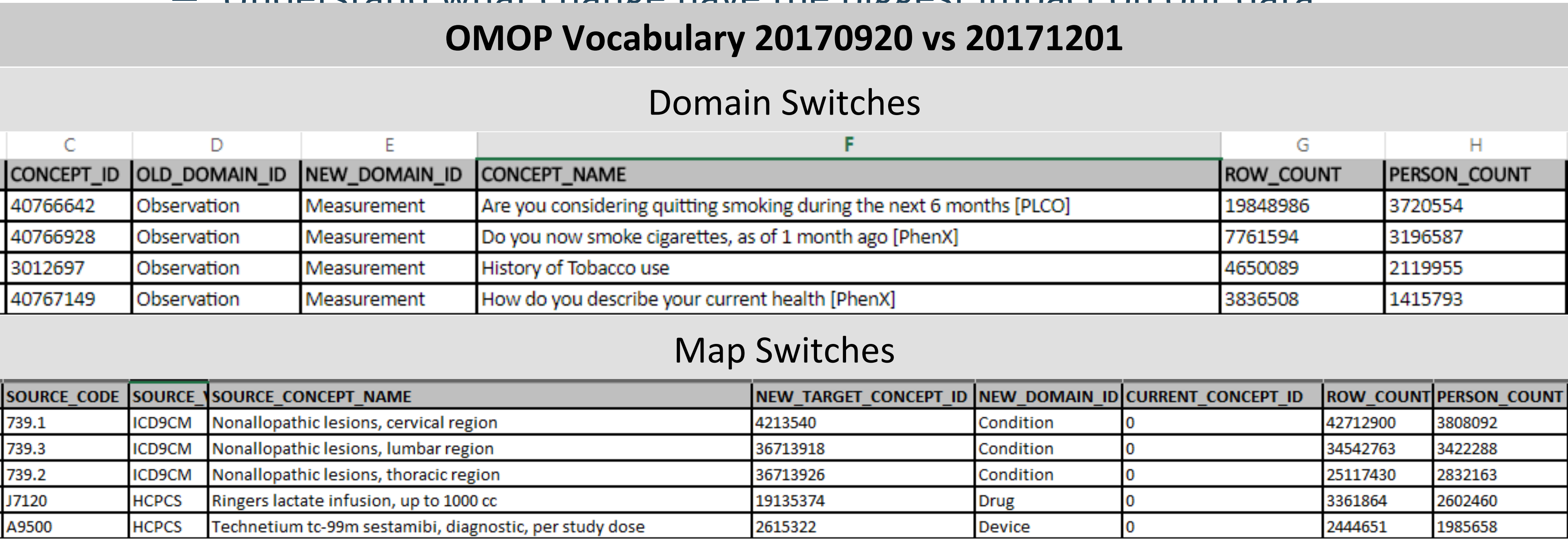
Janssen Bug Fix Sprint Process

1. Select what we can tackle in the month
2. Developers make change / update test cases
3. Test updated Builder
4. Update and test until all test cases pass
5. Run full CDM Build
6. Run ACHILLES, review HEEL
7. Connect to ATLAS, test “Dummy Cohorts”
8. Bless or Reject CDM

Vocabulary Compare

- Your ETL can be working perfectly by adopting a new Vocabulary can bring change

- We try to quantify this change but:
  - Characterizing the differences between two versions of the Vocabulary
  - Understand what change have the biggest impact on our data



OMOP Vocabulary 20170920 vs 20171201

Domain Switches

C	D	E	F	G	H
CONCEPT_ID	OLD_DOMAIN_ID	NEW_DOMAIN_ID	CONCEPT_NAME	ROW_COUNT	PERSON_COUNT
40766642	Observation	Measurement	Are you considering quitting smoking during the next 6 months [PLCO]	19848986	3720554
40766928	Observation	Measurement	Do you now smoke cigarettes, as of 1 month ago [PhenX]	7761594	3196587
3012697	Observation	Measurement	History of Tobacco use	4650089	2119955
40767149	Observation	Measurement	How do you describe your current health [PhenX]	3836508	1415793

Map Switches

SOURCE_CODE	SOURCE_V	SOURCE_CONCEPT_NAME	NEW_TARGET_CONCEPT_ID	NEW_DOMAIN_ID	CURRENT_CONCEPT_ID	ROW_COUNT	PERSON_COUNT
739.1	ICD9CM	Nonallopathic lesions, cervical region	4213540	Condition	0	42712900	3808092
739.3	ICD9CM	Nonallopathic lesions, lumbar region	36713918	Condition	0	34542763	3422288
739.2	ICD9CM	Nonallopathic lesions, thoracic region	36713926	Condition	0	25117430	2832163
J7120	HCPCS	Ringers lactate infusion, up to 1000 cc	19135374	Drug	0	3361864	2602460
A9500	HCPCS	Technetium tc-99m sestamibi, diagnostic, per study dose	2615322	Device	0	2444651	1985658

Janssen Best Practices & Comments

- Write out test cases before programming
  - The list will keep you focus
  - Walk through the ETL document to generate the list
- Generating "dummy test data" has allowed us to test edge cases that may never occur in the data
- Try to test one item at a time
- Have your testing environment set and ready to go
- Some database are not as performant as others with the INSERT.SQL, a non parallel database is preferred (e.g. MS SQL Server)
- This presentation primarily demos how Janssen R&D tests, testing processes could and might look slightly different for your organization

Brought to you by:

Without Martijn Schuemie's contribution of OHDSI tools White Rabbit and Rabbit in a Hat, testing your CDM would be painful. Thank you!

Resources

- White Rabbit GitHub (inclusive of Rabbit In a Hat)
- White Rabbit Wiki
- Rabbit in a Hat Wiki
- Rabbit in a Hat Testing Framework Wiki
- Janssen Truven Test Cases