Distributed health data networks allow participants to maintain ownership over their data and to run queries and analytic programs in the privacy of their own data environments. Some networks, such as the PCORnet Distributed Research Network, utilize machine-generated queries powered by PopMedNet (PMN).

### Background

Machine-generated queries, like those used in PCORnet, can be verbose and difficult for a human to parse. This presents various challenges when testing queries to ensure they perform as expected within different relational database management systems (RDBMS), especially in the fast-paced context of a distributed research network.

### Problem

We developed an open-source Test Case Inserter (TCI) tool and a standard testing process, enabling efficient verification of machine-generated queries with a high degree of confidence without requiring the tester to have any SQL knowledge. This process is not specific to PMN or PCORnet and can be used with any querying application, data model, and supported RDBMS.

### Solution

- Patients are inserted into multiple instances of Oracle, Postgres, and SQL Server via the TCI
- TCI is a Java-based command line application
- TCI ensures the same patients are entered in all RDBMS instances

### Define Queries

- Queries being tested are created in PopMedNet and executed against updated databases
- Some validation queries are modified to stratify on fields used for fake patient identifiers

### Create Test Patients

- We create patients to meet our test cases in an Excel document adhering to the PCORnet Common Data Model
- We assign fake identifiers to each test patient for tracking
- Patents are created to either be included or excluded in query results
- The TCI allows us to create as many patients as appropriate

### Insert Data

- Patients are inserted into multiple instances of Oracle, Postgres, and SQL Server via the TCI
- TCI is a Java-based command line application
- TCI ensures the same patients are entered in all RDBMS instances

### Run Queries

- Results are reviewed against the expected patient results as defined in original test plans
- Queries are given a Pass or Fail verification

### Verify Query Results

- If a query fails, the source data are investigated
- Additional test queries and/or patients are created and SQL is inspected to reproduce and identify issues
- If PMN defects are found, they are logged, fixed, and retested using the same process and test cases in all supported systems

### Currently supported RDBMS Platforms

- Oracle 11, 12
- PCORnet CDM 3.0, 3.1
- Postgres 9.4, 9.5
- PCORnet CDM 3.0, 3.1
- SQL Server 2012, 2014
- PCORnet CDM 3.0, 3.1

### Outcomes

- This testing process has successfully identified unexpected query behavior in various conditions, most of which are complex queries involving joins with patient-record information across several tables, or utilizing different encounter time windows each associated with different patient information within a single query.
- Unexpected use of query input parameters can be a major contributor to errors in resulting query behavior. PopMedNet employs the use of request templates to reduce user error without sacrificing flexibility.

### Future Work

- This methodology will continue to be used to verify PCORnet Menu-Driven Queries in PopMedNet
- Investigate the use of PATID fields and other ways of identifying specific test patients
- Investigate test automation where applicable

Special Thanks: Hozefa Divan, PhD, MSPH for scientific guidance in creating test plans and Zachary Wyner, MPH for poster design.