04 05 2021

Marc Padros

VALUE-Dx WP3 – Task 3.4

POC 1 Infrastructure documentation

VALUE-Dx

This project has received funding from the Innovative Medicines Initiative 2 Joint Undertaking under grant agreement No 820755. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and EFPIA and bioMérieux SA, Janssen Pharmaceutica NV, Accelerate Diagnostics S.L., Abbott, Bio-Rad Laboratories, BD Switzerland Sàrl, and The Wellcome Trust Limited.



University
of Antwerp


BIOMÉRIEUX


wellcome


UNIVERSITY OF
OXFORD


UMC Utrecht


Penta
Child Health Research


UNIVERSITÀ
di VERONA


THE UNIVERSITY
of EDINBURGH


University Medical Center Groningen

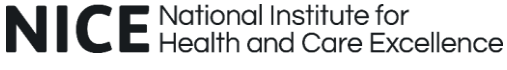

RAMBAM
Health Care Campus


UNIVERSIDAD
DE LA RIOJA


BOSTON
UNIVERSITY


BIOASTER
MICROBIOLOGY TECHNOLOGY INSTITUTE


FIND


NICE National Institute for
Health and Care Excellence


Gesundheit Österreich
GmbH


ESCMID
EUROPEAN SOCIETY
OF CLINICAL MICROBIOLOGY
AND INFECTIOUS DISEASES


ERS
EUROPEAN
RESPIRATORY
SOCIETY
every breath counts


Berry Consultants
Statistical Innovation


IBBL
INTEGRATED BIOBANK OF LUXEMBOURG
FOR NEXT GENERATION HEALTHCARE


ZonMw


janssen


ACCELERATE
DIAGNOSTICS


Abbott


BIO-RAD


BD

Table of contents

Federated services

1.1. Arachne Research Network

The OHDSi's Arachne research network is a software suite enabling researchers to conduct federated collaborative research studies across healthcare organizations, data owners and researches. Contrary to traditional research architecture where research data is stored and analysed centrally, data within a federated system remains local. Analysis queries are designed and distributed centrally, however query execution is handled remotely. Federated data analysis platforms rely on harmonised data platforms or common data models such as OMOP CDM. We have leveraged the OHDSI platform in this POC to set up a federated network for AMR data from different sources. The main advantage of using federated data analysis systems is that interoperability and data analysis of isolate level susceptibility testing data can be orchestrated

across an international network of participating laboratories, without the need for individual laboratories to share this sensitive patient identifiable data - all that is shared is the query and the aggregated result.

The Arachne suite consists of two main components, Arachne Central and Arachne Data Node.

1.1.1. Arachne Central

The Arachne Central (AC) component is the main UI where studies and analysis can be set-up and distributed across participating data nodes. After code execution on one or more nodes, results are uploaded into the AC and made available for publishing or further processing.

The query distribution and execution process can be automated with AC enterprise edition, but this edition has been left out of scope for this current POC.

1.1.2. Arachne Data Node

The Arachne Data Node component facilitates connection and communication between local databases and Arachne Central. It provides a user interface (UI) for OMOP CDM data sources and the possibility to execute SQL and R code on the local data sources. The data node has two modes of operation, Standalone and network mode. Data nodes configured in network mode are connected to an Arachne Central instance. Standalone data nodes can operate without Arachne central. Data nodes uses the Execution Engine module to execute SQL or R code against the actual database(s). The Execution Engine leverages SQLRender (https://ohdsi.github.io/SqlRender/) to render parametrized SQL and translate to different SQL dialects (e.g. ohdsi sql (parametrized) to MSSQL).

2. POC 1

2.1. Design

POC 1 consists of one instance of AC (Observatory) and two data nodes + EE combo's (BioMérieux and Find) (Figure 1).

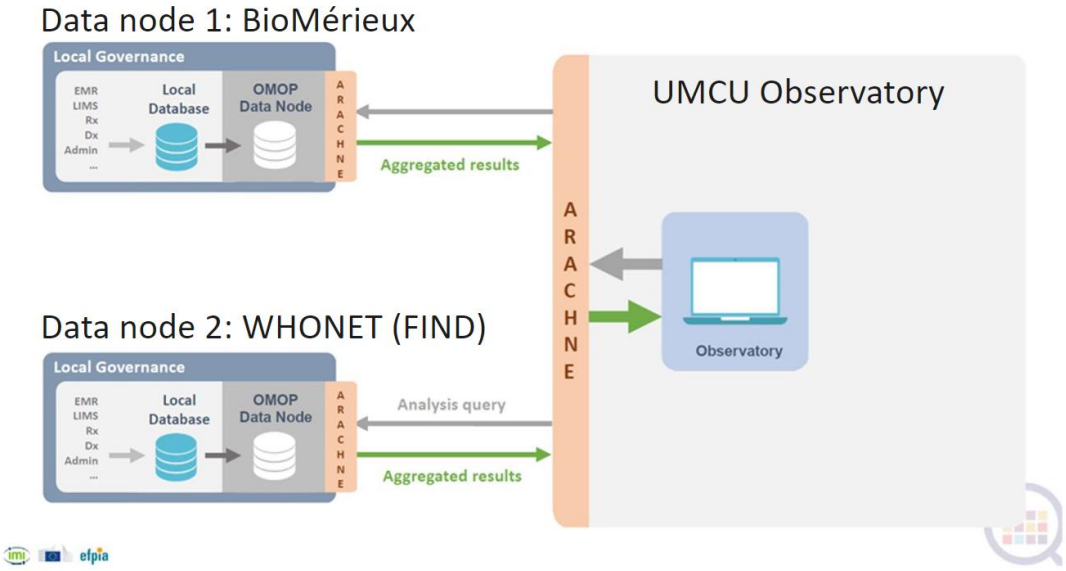


Figure 1: POC1 architecture

2.1.1. Setting up the Observatory

The observatory consists of an instance of Arachne Central and a data node containing mock data. This data can be used by data analysts and researchers to design and test new queries before distributing them over the network. The two steps to setup up the observatory are as follows:

1) Setting up Arachne Central instance (Setup guide)
2) Setting up a local data node (see 2.2.2.)

2.1.2. Setting up local data nodes (BioMérieux, Find)

Setting up a local data node requires a stepped approach which will be described below.

1) Setting up a database instance (e.g., MS SQL, PostgreSQL, MySQL)
2) Create an instance of the OMOP common data model on the database by following the steps as shared by the OHDSI community (https://github.com/OHDSI/CommonDataModel)
3) Load cdm vocabulary (VocabImport)
4) Load data into CDM (CDM import script)
5) Setup Data Node in NETWORK mode (Setup guide)

6) Setup EE (Setup guide)
7) Configure data source in data node (Figure 2Figure 3)

Data node 1 was hosted on an instance of MS SQL server and was filled with data as provided by BioMérieux. Data node 2 was hosted on an instance of PostgreSQL and filled with data from WHONET.

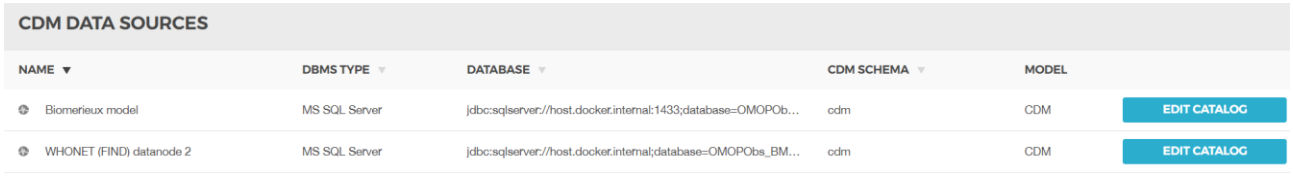


Figure 2: Data source configuration in data node

2.1.3. Setting up connectivity to the Observatory

The local data node should be configured to run in 'Network mode' to enable communication from and to the Observatory. This is done by setting the 'datanode.runMode' parameter to 'Network'. Standalone configurations will not be able to, for example, publish data sources to the Observatory or add new users from the Observatory. Once the local data node has been properly setup and running, the Observatories details need to be configured in the system-settings. After correct configuration, the local node should be able to communicate with the Observatory. (Figure 3)

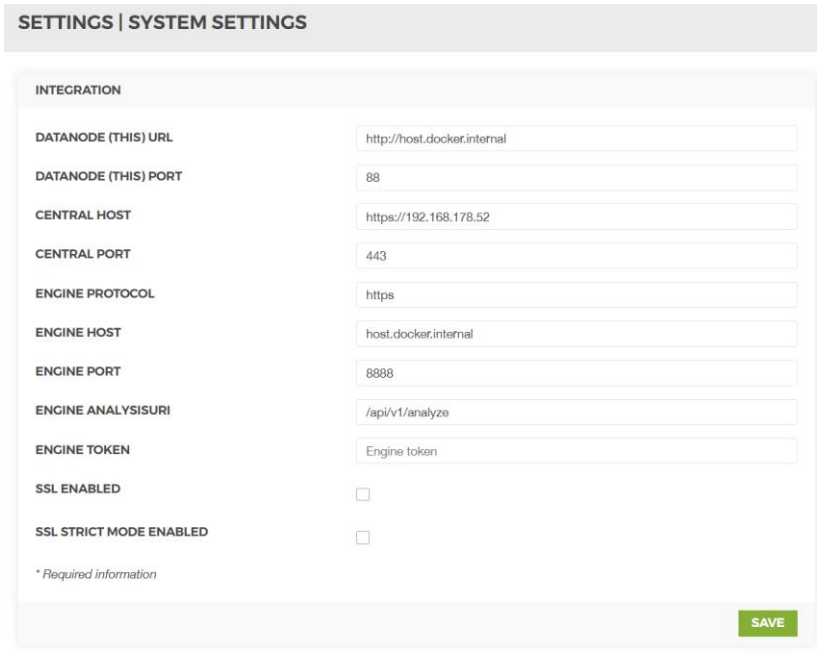


Figure 3: Configuring system settings for data node

2.1.4. Use case definition

Lab-Select is a visionary service defined as part of the task 3.4 work which describes a pan European service allowing 'research organisations' to identify and select laboratories to

collaborate with in AMR studies. The Lab-Select concept serves two purposes, firstly it provides a use case for the Task 3.4 investigation into data sharing architectures, and secondly it provides information and thinking to input into ECRIN/ECRAID initiatives (to establish a sustainable Clinical Trial Network).

An initial set of use cases are described in the product design specification and requirements document on the Value-Dx project site (VALUE-Dx Lab Select PDR and PDS v1.0.XLSX)

2.2. Lab-Select Characteristics

This section lists major Lab-Select characteristics.

2.2.1. Target Setting

Lab-Select is a Pan European service allowing individual laboratories conducting AMR testing to enrol and European researchers to apply for access.

2.2.2. Target Users

Lab-Select is targeted primarily at research organisations such as pharmaceutical companies who are planning to conduct drug discover investigations or clinical trials for new drug treatments. It can also be used by other academic research institutions or research programmes such as Value-Dx.

This simple 'EPIC' requirement demands a number of fundamental entities and components be in place, they are as follows:

1 – A Centralised provider to delivery of the service, known here as the Lab-select Observatory

2 – A Network of participating laboratories – known as the Lab-Select network

3 – Individual laboratories generating fully characterised AMR data case level data – known later in this document as data as from each laboratory

4 – Supporting 'inventory' data describing the laboratory itself

2.2.3. Service Proposition

The Lab-Select service proposition is to provide the researcher with the ability to;

- Select laboratories from the network
- Contact the laboratories and agree collaborations
- Gain access to the collective case-level data of the collaboration laboratories
- Provide the ability to conduct research activities (data analytics) on real world data

This 'User journey' demands the following additional features:

- Inventory search – Search and select base on matching inventory e.g. location, QA standards used, available diagnostics

- Surveillance search and testing profiles – search and select based on trending and profiles data i.e. prevalence of pathogens regularly tested for, samples handled, antibiogram. This is akin to maintaining surveillance type data for each lab
- Bespoke analytics – bespoke, researcher defined case-level analysis data mining, across all selected laboratories, in order to either further refine selection or as part of conducting the required research. The analysis can span years and by its nature is multi-site

2.3. Implementation

The implementation of POC 1 can be separated into three phases:

1) Study design
2) Study execution
3) Results dissemination and visualisation

Figure 4 represents the full analysis workflow combining the three phases.

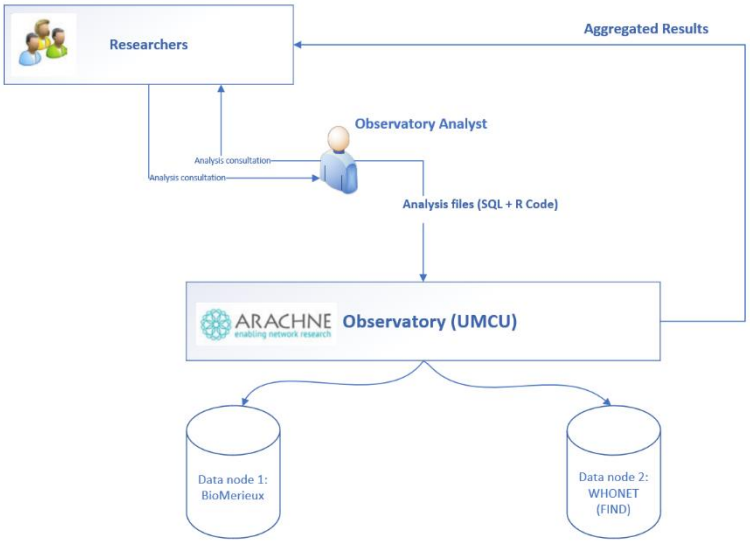


Figure 4: Analysis Workflow

2.3.1. Phase 1: Study design

This phase is started with a study question that is formulated by a study team. When a study question is deemed feasible, the analysis workflow is started. This flow begins with the researchers contacting the observatory analyst for a new project. The analyst then translates the research question(s) into SQL and/or R based analysis files. These files can be executed against the observatory's data node to test and finetune. When ready, the study moves to phase 2, study execution.

2.3.2. Phase 2: Study execution

After completion of phase 1, the analysis files are ready to be distributed across the participating data nodes. In order to do so, a new study is created on the Observatory. (Figure 5, Figure 6)

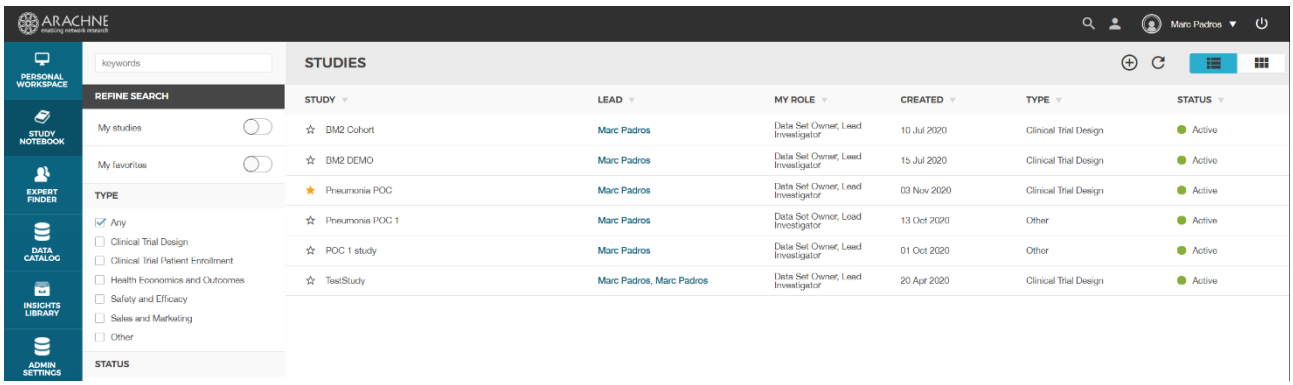


Figure 5: Study overview

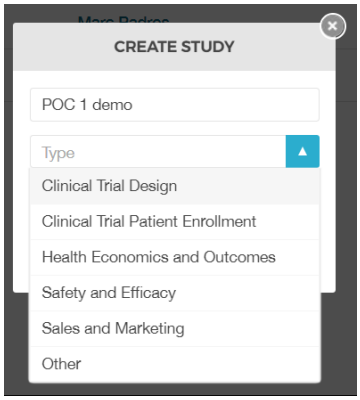


Figure 6: Creating a new study

After the new study is created, the study dashboard becomes visible making it possible to enter or edit the studies metadata. The two main sections for POC 1 are 'Data sources' and Analyses. In data sources, the relevant data sources for the study are selected. (

)

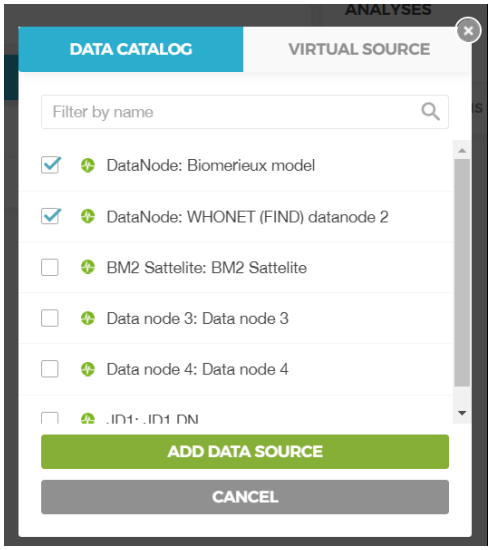


Figure 7: Selecting participating data sources

By creating a new analysis, the researchers can upload the analysis files and submit them to the participating data sources. (Figure 8) Data source owners receive a notification after code

submission to their node and are asked to execute the code against their data source. After code execution, the results are loaded back into the observatory, finalising phase 2. (Figure 9)

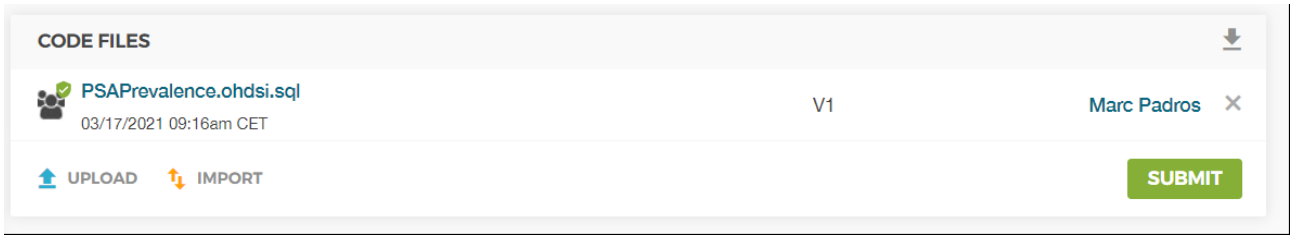


Figure 8: Submitting code to data sources

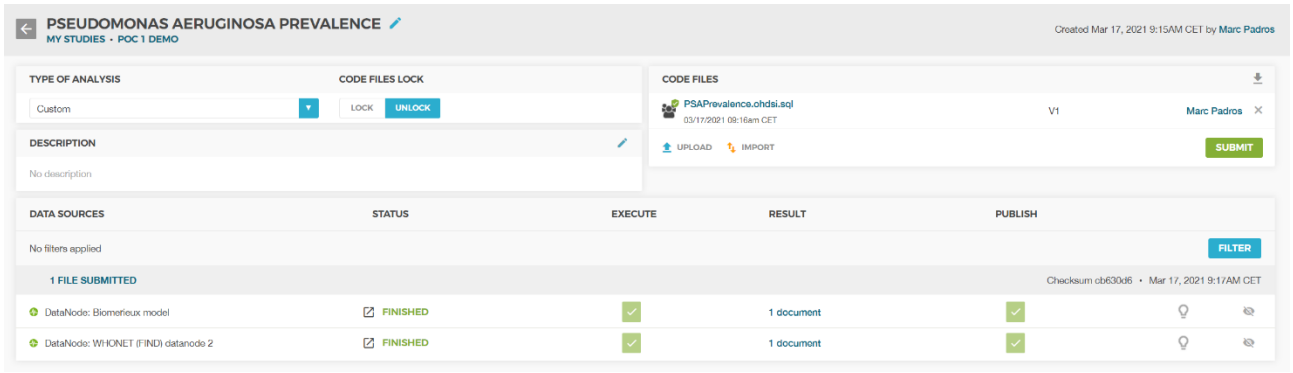


Figure 9: Finalised code execution

2.3.3. Phase 3: Results dissemination and visualisation

In the final phase of a study execution, all results generated by participating data sources are extracted from the portal and merged into a final analysis data set. This final data set is then made available to the researchers and can also be used for exploratory data analysis (e.g. by visualisation). (Figure 10)

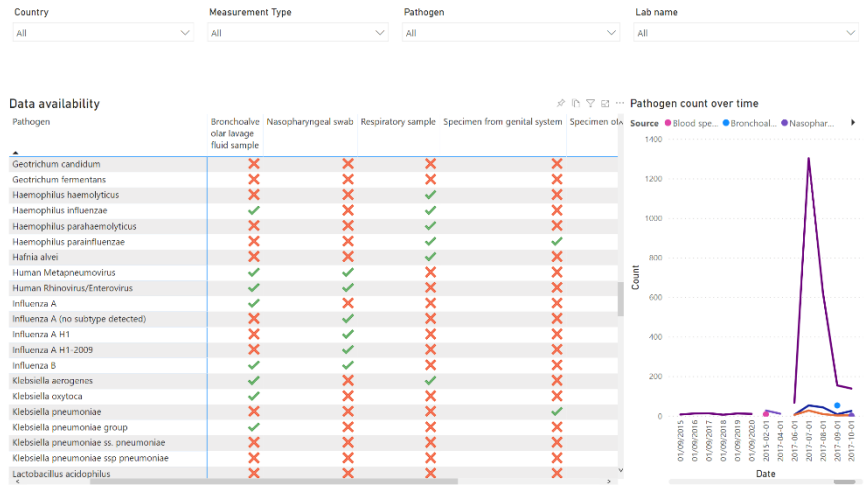


Figure 10: Data visualisation

3. Findings

3.1. Handling several SQL dialects (e.g., MySQL, PostgreSQL, MSSQL)

The Arachne software stack leverages the SQLRender r-package to translate parametrized SQL queries into several dialects. Within this POC we used a mix of MSSQL and PostgreSQL data nodes. One of the issues we experienced when developing queries at the observatory level is that not all transact SQL functions are being handled by SQLRender. This results in a somewhat limited set of functions that are available when developing custom queries, potentially impacting SQL execution performance.

3.2. Vulnerabilities associated with custom queries

When creating analysis code within the Observatory, several analysis types can be selected. All types, except 'Custom', are handled by Arachne's code and produce standardised outcome formats for each analysis type. This is a secure design as the operator cannot interfere with the actual code that is being executed against the data node. As a result, only aggregated results will be shared with the Observatory. For 'Custom' types however, this does not apply. Arachne will not automatically prevent custom queries from returning record level results, if so designed. So, although custom queries provide endless options for analysis, attention should be given to the possibility of disclosing unwanted levels of details.

3.3. Process automation

For POC 1 we've implemented the community edition of both Arachne Central (Observatory) and Arachne Data Node (local data nodes). This edition does not implement the ability to distribute queries to- and gather results from- participating data nodes automatically. The process of query distribution and data collection is therefore quite manual. For a custom study approach, this may not be a big issue as data owners will likely want to have full control on all steps involved in the study. However, for a surveillance or Lab Select setting, where the same query is executed at a certain time interval, automation of this process may be necessary to keep data nodes involved. Automation can be achieved by using the Arachne Enterprise Edition or investigating whether the current API could provide this functionality.

4. Addendum

4.1. CDM import script

```
BULK INSERT cdm.Person
FROM '[PATHTOFile]\patients.csv'
WITH ( FIRSTROW=2
      , FIELDTERMINATOR = ','
      , ROWTERMINATOR = '0x0a');


BULK INSERT cdm.care_site
FROM '[PATHTOFile]\careSites.csv'
WITH ( FIRSTROW=2
      , FIELDTERMINATOR = ','
      , ROWTERMINATOR = '0x0a');



BULK INSERT cdm.measurement1
FROM '[PATHTOFile]\measurements2.csv'
WITH (

CODEPAGE = '65001',
DATAFILETYPE ='char',
FIRSTROW=2
      , FIELDTERMINATOR = ','

      , ROWTERMINATOR = '0x0a');




GO

INSERT INTO
       cdm.measurement (
          [measurement_id]
      ,[person_id]
      ,[measurement_concept_id]
      ,[measurement_date]
      ,[measurement_datetime]
      ,[measurement_time]
      ,[measurement_type_concept_id]
      ,[operator_concept_id]
      ,[value_as_number]
      ,[value_as_concept_id]
      ,[unit_concept_id]
      ,[range_low]
      ,[range_high]
      ,[provider_id]
      ,[visit_occurrence_id]
      ,[visit_detail_id]
      ,[measurement_source_value]
      ,[measurement_source_concept_id]
      ,[unit_source_value]
      ,[value_source_value])
select [measurement_id]
      ,[person_id]
      ,[measurement_concept_id]
      ,[measurement_date]
```

```
      ,[measurement_datetime]
      ,[measurement_time]
      ,[measurement_type_concept_id]
      ,[operator_concept_id]
      ,[value_as_number]
      ,[value_as_concept_id]
      ,[unit_concept_id]
      ,[range_low]
      ,[range_high]
      ,[provider_id]
      ,[visit_occurrence_id]
      ,[visit_detail_id]
      ,[measurement_source_value]
      ,c.[concept_id] -- OMOP concept id
      ,[unit_source_value]
      ,[value_source_value]
FROM
       cdm.measurement1 t1 inner join cdm.concept c ON t1.measurement_source_concept_id =
c.concept_code

delete from cdm.measurement1

BULK INSERT cdm.observation
FROM '[PATHTOFile]\observations2.csv'
WITH (

CODEPAGE = '65001',
DATAFILETYPE ='char',
FIRSTROW=2
      , FIELDTERMINATOR = ','

      , ROWTERMINATOR = '0x0a');



BULK INSERT cdm.fact_relationship
FROM '[PATHTOFile]\relationships2.csv'
WITH (

CODEPAGE = '65001',
DATAFILETYPE ='char',
FIRSTROW=2
      , FIELDTERMINATOR = ','

      , ROWTERMINATOR = '0x0a');



BULK INSERT cdm.specimen
FROM '[PATHTOFile]\specimens.csv'
WITH (

CODEPAGE = '65001',
DATAFILETYPE ='char',
FIRSTROW=2
      , FIELDTERMINATOR = ','

      , ROWTERMINATOR = '0x0a');



BULK INSERT cdm.visit_occurrence
FROM '[PATHTOFile]\visitOccurence.csv'
```

```
WITH (

CODEPAGE = '65001',
DATAFILETYPE ='char',
FIRSTROW=2
      , FIELDTERMINATOR = ','

      , ROWTERMINATOR = '0x0a');


CREATE SEQUENCE op1
START WITH 1
INCREMENT BY 1 ;

insert into cdm.observation_period (observation_period_id,person_id,
observation_period_start_date, observation_period_end_date, period_type_concept_id)
select next value for op1, person_id, '1990-01-01' startdate, '2050-01-01' enddate, 44814722
periodconceptid from cdm.person
```

4.2. POC 1 Data analysis queries

```
DROP TABLE

IF EXISTS ##q1
       SELECT p.person_id
              ,p.concept_name Gender
              ,CASE
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) BETWEEN 0
                                   AND 10
                            THEN '0-10'
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) BETWEEN 10
                                   AND 20
                            THEN '10-20'
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) BETWEEN 20
                                   AND 30
                            THEN '20-30'
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) BETWEEN 30
                                   AND 40
                            THEN '30-40'
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) BETWEEN 40
                                   AND 50
                            THEN '40-50'
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) BETWEEN 50
                                   AND 60
                            THEN '50-60'
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) BETWEEN 60
                                   AND 70
                            THEN '60-70'
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) BETWEEN 70
                                   AND 80
                            THEN '70-80'
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) BETWEEN 80
                                   AND 90
                            THEN '80-90'
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) BETWEEN 90
                                   AND 100
                            THEN '90-100'
                     WHEN DATEDIFF(YEAR, p.birth_datetime, m.measurement_date) > 100
                            THEN '>100'
                     ELSE NULL
                     END AS AgeGroupatMeasurement
              ,DATEFROMPARTS(YEAR(s.specimen_date), MONTH(s.specimen_date), 1) SpecimenDate
              ,p.address_1 address
              ,p.city
```

```
	,p.county Country --used for demo
	,s.specimen_id
	,s.SpecimenSource SpecimenSource
	,m.measurement_source_value m_measurement_source_value
	,m.concept_name MeasurementConceptName
	,o.observation_source_value Pathogen
	,o.observation_source_concept_id
	,o.observation_id
	,m2.measurement_source_value m2_measurement_source_value
	,m2.measurement_source_concept_id
	,m2.value_source_value
	,m2.measurement_id
	,crcomp.concept_name ComponentName
	,CASE
		WHEN crmethod.concept_name IS NOT NULL
			THEN crmethod.concept_name
		ELSE crprop.concept_name
		END MeasurementMethod
	,CASE
		WHEN crmethod.concept_id IS NOT NULL
			THEN crmethod.concept_id
		ELSE crprop.concept_id
		END MethodConceptID
	,lab.labname
INTO ##q1
FROM (
	SELECT p.*
		,co.concept_name
		,c.care_site_name
		,c.care_site_source_value
		,c.place_of_service_concept_id
		,c.place_of_service_source_value
		,l.address_1
		,l.city
		,l.county
	FROM @cdm_database_schema.person p
	INNER JOIN @cdm_database_schema.care_site c ON p.care_site_id = c.care_site_id
	INNER JOIN @cdm_database_schema.location l ON c.location_id = l.location_id
	LEFT JOIN @cdm_database_schema.concept co ON p.gender_concept_id =
	co.concept_id
	) p
		--Person combined with care_site and location
INNER JOIN (
	SELECT s.*
		,c.concept_name SpecimenSource
	FROM @cdm_database_schema.specimen s
	INNER JOIN @cdm_database_schema.concept c ON s.specimen_concept_id =
	c.concept_id
	) s ON p.person_id = s.person_id
		--Join Specimens
INNER JOIN (
	SELECT *
	FROM @cdm_database_schema.fact_relationship
	WHERE relationship_concept_id = 32669
	) s_fr ON s.specimen_id = s_fr.fact_id_1
		--Join Specimen to fact_relationship
INNER JOIN (
	SELECT m.*
		,c.concept_class_id
		,concept_name
	FROM @cdm_database_schema.measurement m
	INNER JOIN @cdm_database_schema.concept_relationship cr ON
	m.measurement_concept_id = cr.concept_id_1
	INNER JOIN @cdm_database_schema.concept c ON cr.concept_id_2 = c.concept_id
	WHERE c.concept_class_id = 'LOINC Method'
```

```
        ) m ON s_fr.fact_id_2 = m.measurement_id
                --Join fact_relationship to Measurement 1
    LEFT JOIN (
        SELECT *
        FROM @cdm_database_schema.fact_relationship
        WHERE relationship_concept_id = 581411
        ) o_fr ON m.measurement_id = o_fr.fact_id_1
                --Join Measurement 1 to fact_relationship
    LEFT JOIN @cdm_database_schema.observation o ON o_fr.fact_id_2 = o.observation_id
                --Join fact_relationship to Observation 1
    INNER JOIN (
        SELECT *
        FROM @cdm_database_schema.fact_relationship
        WHERE relationship_concept_id = 581410
        ) om_fr ON o.observation_id = om_fr.fact_id_1
                --Join Observation 1 to fact_relationship
    LEFT JOIN (
        SELECT *
        FROM @cdm_database_schema.measurement
        WHERE measurement_concept_id <> 3044054
        ) m2 ON om_fr.fact_id_2 = m2.measurement_id
                --Join fact_relationship to Measurement 2 (and 3, 4, etc)
    LEFT JOIN (
        SELECT r.concept_id_1
                ,c.*
        FROM @cdm_database_schema.concept_relationship r
        INNER JOIN @cdm_database_schema.concept c ON r.concept_id_2 = c.concept_id
        WHERE relationship_id = 'Has component'
                AND vocabulary_id = 'LOINC'
                AND domain_id = 'Observation'
        ) crcomp ON m2.measurement_concept_id = crcomp.concept_id_1
        --Add metadata to Measurement 2
    LEFT JOIN (
        SELECT r.concept_id_1
                ,c.*
        FROM @cdm_database_schema.concept_relationship r
        INNER JOIN @cdm_database_schema.concept c ON r.concept_id_2 = c.concept_id
        WHERE relationship_id = 'Has method'
                AND vocabulary_id = 'LOINC'
                AND domain_id = 'Observation'
        ) crmethod ON m2.measurement_concept_id = crmethod.concept_id_1
        --Add metadata to Measurement 2
    LEFT JOIN (
        SELECT r.concept_id_1
                ,c.*
        FROM @cdm_database_schema.concept_relationship r
        INNER JOIN @cdm_database_schema.concept c ON r.concept_id_2 = c.concept_id
        WHERE relationship_id = 'Has property'
                AND vocabulary_id = 'LOINC'
                AND domain_id = 'Observation'
        ) crprop ON m2.measurement_concept_id = crprop.concept_id_1
        --Add metadata to Measurement 2
    CROSS JOIN (
        SELECT cdm_source_name labname
        FROM @cdm_database_schema.cdm_source
        ) lab
        --Join labinfo to results table
    ORDER BY p.person_id
        ,m.measurement_id



/*********************************************************/
```

```
/*                          Q1                          */
/******************************************************/
--Method count by Pathogen, source, SpecimenDate,
MeasurementConceptDate, Labname and country
SELECT Pathogen
       ,SpecimenDate
       ,SpecimenSource
       ,Country
       ,MeasurementConceptName
       ,Labname
       ,count(pathogen) pathogencount
FROM (
       SELECT DISTINCT SpecimenDate
              ,pathogen
              ,specimensource
              ,country
              ,observation_id
              ,MeasurementConceptName
              ,labname
       FROM ##q1
       ) q
GROUP BY Pathogen
       ,SpecimenDate
       ,SpecimenSource
       ,Country
       ,MeasurementConceptName
       ,labname
ORDER BY Pathogen



/******************************************************/
/*                          Q2                          */
/******************************************************/
--Method count by Pathogen, source and country
SELECT
       Pathogen
       ,SpecimenSource
       ,Country
       ,SpecimenDate
       ,ComponentName
       ,value_source_value
       ,m_measurement_source_value
       ,count(MeasurementMethod) MethodCount
FROM ##q1
WHERE measurementconceptname = 'Culture'
       AND MethodConceptID = 1029735
GROUP BY
       Pathogen
       ,SpecimenSource
       ,Country
       ,ComponentName
       ,value_source_value
       ,SpecimenDate
       ,m_measurement_source_value
```

www.imi.europa.eu www.value-dx.eu