Reporting in OpenMRS
A MambaETL Showcase

Vincent Yahuma, UCSF Kenya
Arthur Mugume, Makerere University School of Public Health
Samuel Lubwama, Makerere University School of Public Health
Derrick Baluku, Makerere University School of Public Health
Florentin Ndabarasa, Rwanda Ministry of Health

UCSF

MAKERERE UNIVERSITY

REPUBLIC OF RWANDA
MINISTRY OF HEALTH

MambaETL

PEPFAR

TAP

Why MambaETL?

PEPFAR

Technical Assistance Platform

The Problem:

Reporting systems such as Excel, JasperReports, SuperSet, Chart.JS or PowerBI are designed to work with wide tables

But...
OpenMRS does not support this out of the box, making reporting complex and slow

PEPFAR

Technical Assistance Platform

Transposing data

MambaETL solves this problem by converting the data from long to wide format

"Long" format

country	year	metric
x	1960	10
x	1970	13
x	2010	15
y	1960	20
y	1970	23
y	2010	25
z	1960	30
z	1970	33
z	2010	35

"Wide" format

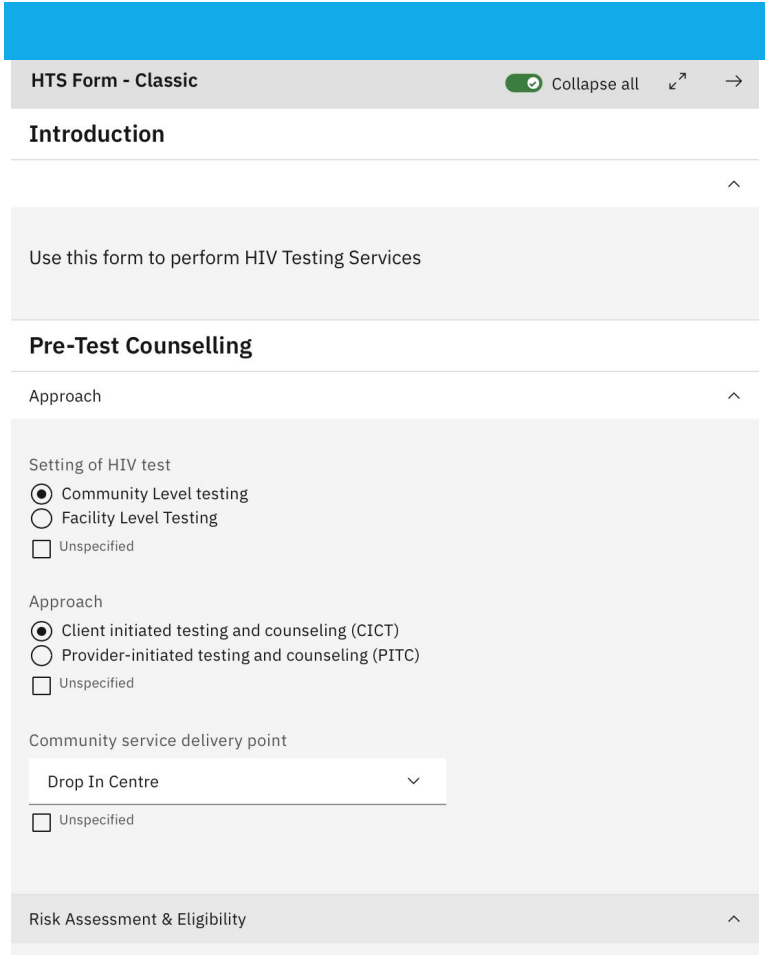
country	yr1960	yr1970	yr2010
x	10	13	15
y	20	23	25
z	30	33	35

PEPFAR Technical Assistance Platform

Basic data collection in OpenMRS

In OpenMRS, most data is collected through a form

The form creates an encounter and several observations in the database



An HTS Form

- OpenMRS stores patient observational data in a long format
- For each encounter, such as HTS, multiple tables get records added and multiple rows are added into the the Obs table, often exceeding +30 rows per encounter
- These tables can rapidly expand, especially considering the number of patients and encounters recorded in a health facility

PEPFAR

Technical Assistance Platform

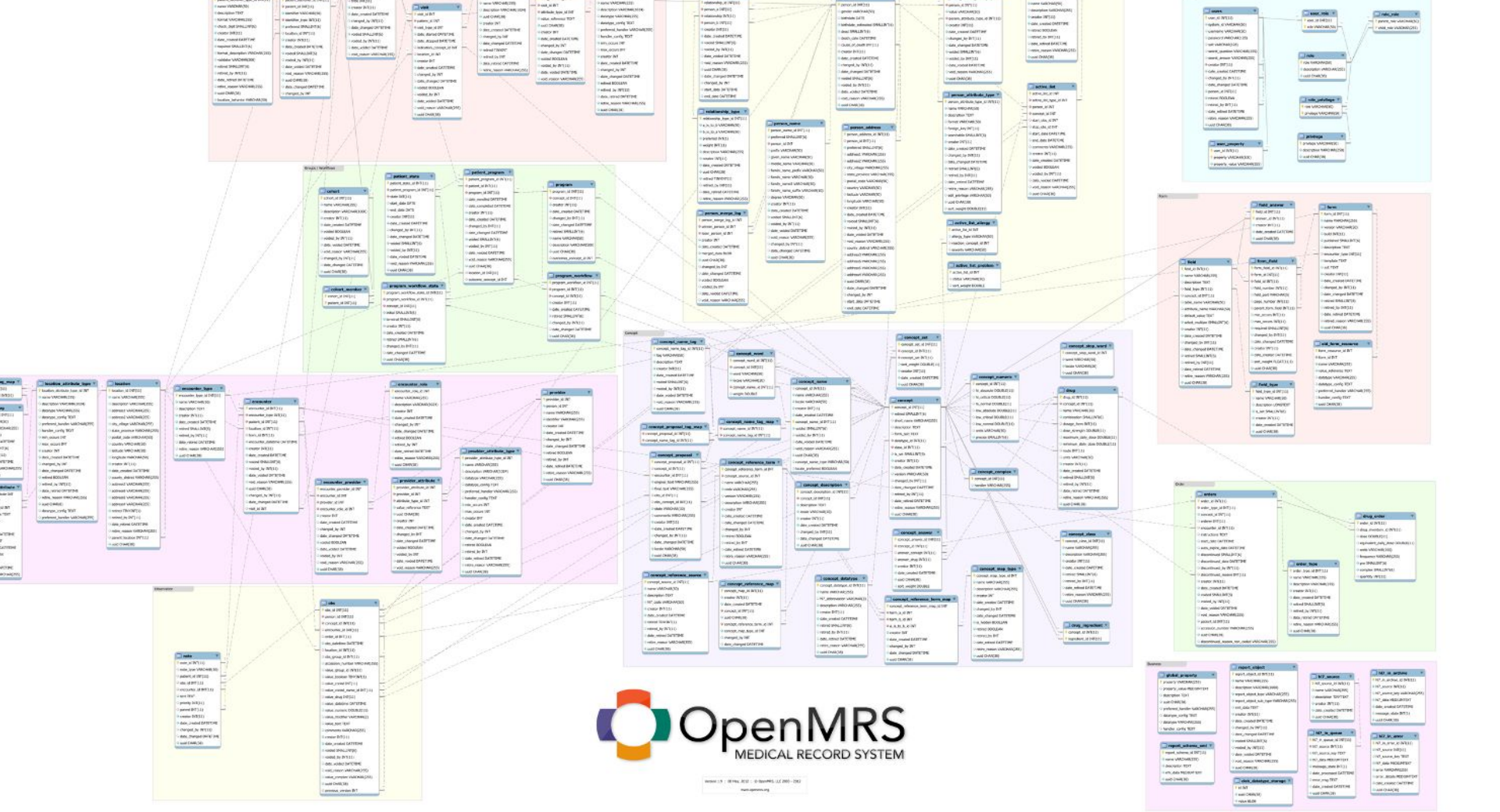
Observations

Here the patient (patient_id 205) has 35 rows in the obs table to represent the hts encounter for that day

	obs_id	person_id	concept_id	encounter_id	obs_datetime	value_coded	value_datetime
1	675578	205	929	2523	2022-11-15 00:00:00	1826	<null>
2	677378	205	104	2523	2022-11-15 00:00:00	106	<null>
3	678295	205	1000	2523	2022-11-15 00:00:00	53	<null>
4	678585	205	120	2523	2022-11-15 00:00:00	124	<null>
5	679555	205	945	2523	2022-11-15 00:00:00	76	<null>
6	680514	205	942	2523	2022-11-15 00:00:00	76	<null>
7	682044	205	69	2523	2022-11-15 00:00:00	1980	<null>
8	684637	205	109	2523	2022-11-15 00:00:00	32	<null>
9	685240	205	939	2523	2022-11-15 00:00:00	1086	<null>
10	685598	205	1002	2523	2022-11-15 00:00:00	89	<null>
11	686479	205	48	2523	2022-11-15 00:00:00	88	<null>
12	687208	205	48	2523	2022-11-15 00:00:00	88	<null>
13	689794	205	16	2523	2022-11-15 00:00:00	88	<null>
14	691140	205	102	2523	2022-11-15 00:00:00	88	<null>
15	691685	205	10	2523	2022-11-15 00:00:00	88	<null>
16	692190	205	935	2523	2022-11-15 00:00:00	<null>	<null>
17	693151	205	928	2523	2022-11-15 00:00:00	<null>	<null>
18	693695	205	1012	2523	2022-11-15 00:00:00	<null>	<null>
19	694419	205	954	2523	2022-11-15 00:00:00	<null>	<null>
20	695513	205	1005	2523	2022-11-15 00:00:00	<null>	<null>
21	697177	205	19	2523	2022-11-15 00:00:00	<null>	2022-11-15 00:00:00
22	698138	205	35	2523	2022-11-15 00:00:00	<null>	2022-11-15 00:00:00
23	699138	205	1490	2523	2022-11-15 00:00:00	<null>	<null>
24	700128	205	11	2523	2022-11-15 00:00:00	<null>	<null>
25	701204	205	119	2523	2022-11-15 00:00:00	<null>	<null>
26	702334	205	1013	2523	2022-11-15 00:00:00	<null>	<null>
27	703045	205	13	2523	2022-11-15 00:00:00	<null>	<null>
28	703670	205	955	2523	2022-11-15 00:00:00	<null>	<null>
29	704590	205	997	2523	2022-11-15 00:00:00	<null>	<null>
30	704861	205	998	2523	2022-11-15 00:00:00	<null>	<null>
31	705776	205	927	2523	2022-11-15 00:00:00	<null>	<null>
32	706756	205	943	2523	2022-11-15 00:00:00	<null>	<null>
33	707833	205	98	2523	2022-11-15 00:00:00	<null>	<null>
34	708718	205	1796	2523	2022-11-15 00:00:00	<null>	<null>
35	709564	205	1001	2523	2022-11-15 00:00:00	<null>	<null>

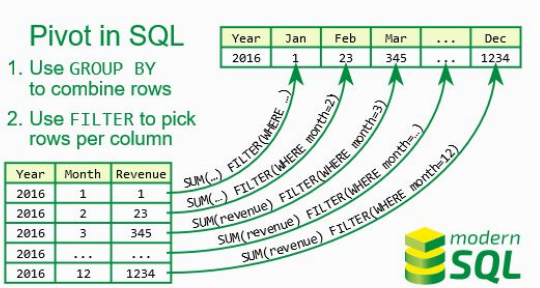
Example of a “long” observation table

PEPFAR TAP


OpenMRS
MEDICAL RECORD SYSTEM

Transposing data

Transposing data manually in SQL is a laborious, error prone and complex job



To make the query more literate, the **`extract`** expression can be moved to a central location. That could be a generated column or a view so that other queries could reuse this expressions. For this example, it is sufficient to centralize the **`extract`** expression within the query—either using the **`with`** clause, or as an inline view:

```
SELECT year
     , SUM(revenue) FILTER (WHERE month =  1) jan_revenue
     , SUM(revenue) FILTER (WHERE month =  2) feb_revenue
     ...
     , SUM(revenue) FILTER (WHERE month = 12) dec_revenue
  FROM (SELECT invoices.*
             , EXTRACT(YEAR  FROM invoice_date) year
             , EXTRACT(MONTH FROM invoice_date) month
          FROM invoices
       ) invoices
 GROUP BY year
```

Transposing data using SQL

PEPFAR TAP

```
CREATE PROCEDURE sp_mamba_flat_encounter_table_insert(
    IN flat_encounter_table_name CHAR(255) CHARACTER SET UTF8MB4
)
BEGIN

    SET session group_concat_max_len = 20000;
    SET @tbl_name = flat_encounter_table_name;

    SET @old_sql = (SELECT GROUP_CONCAT(COLUMN_NAME SEPARATOR ', ')
                    FROM INFORMATION_SCHEMA.COLUMNS
                    WHERE TABLE_NAME = @tbl_name
                      AND TABLE_SCHEMA = Database());

    SELECT
        GROUP_CONCAT(DISTINCT
            CONCAT(' MAX(CASE WHEN column_label = ''', column_label, ''' THEN ',
                fn_mamba_get_obs_value_column(concept_datatype), ' END) ', column_label)
            ORDER BY id ASC)
    INTO @column_labels
    FROM mamba_dim_concept_metadata
    WHERE flat_table_name = @tbl_name;

    SET @insert_stmt = CONCAT(
            'INSERT INTO `', @tbl_name, '` SELECT eo.encounter_id, eo.person_id, eo.encounter_datetime, ',
            @column_labels, '
            FROM mamba_z_encounter_obs eo
                INNER JOIN mamba_dim_concept_metadata cm
                ON IF(cm.concept_answer_obs=1, cm.concept_uuid=eo.obs_value_coded_uuid, cm.concept_uuid=eo.obs_question_uuid)
            WHERE cm.flat_table_name = ''', @tbl_name, '''
            AND eo.encounter_type_uuid = cm.encounter_type_uuid
            GROUP BY eo.encounter_id, eo.person_id, eo.encounter_datetime;');

    PREPARE inserttbl FROM @insert_stmt;
    EXECUTE inserttbl;
    DEALLOCATE PREPARE inserttbl;

END //
```

An example of a transposing query

PEPFAR Technical Assistance Platform

How MambaETL helps

- Does this automatically
- Minimal or no intervention from the implementer/dev (deploy and run)

- MambaETL automatically transposes the data—i.e. converting from long to wide
- In a wide format, each encounter gets one row, and each observation becomes a column
- Stores the transposed data (persisted) for fast querying and processing
- This makes reporting fast, accurate and easy to do

PEPFAR TAP

A flattened table

For the patient_id 205 from the long format the 35 rows have been flattened to 1 row

client_id	encounter_id	encounter_date	test_setting	hts_approach	ever_teste...	consent...	reason_for_test	final_test_result
205	2523	2022-11-15	Facility Level Testi…	Provider-initiated HIV t…	No	Yes	Index Client Testing	Negative

client_id	encounter_id	encounter_date	test_setting	hts_approach	ever_teste...	consent...	reason_for_test	final_test_result
205	2523	2022-11-15	Facility Level Testi…	Provider-initiated HIV t…	No	Yes	Index Client Testing	Negative
206	2524	2023-02-21	Facility Level Testi…	Provider-initiated HIV t…	No	Yes		Positive
207	2525	2023-02-06	Facility Level Testi…	Client Initiated Testing…	No	Yes	Index Client Testing	Negative
208	2526	2023-02-06	Facility Level Testi…	Provider-initiated HIV t…	No	Yes	Index Client Testing	Positive
209	2527	2022-12-15	Facility Level Testi…	Client Initiated Testing…	Yes	Yes	Assisted Partner Not…	Negative
210	2528	2023-01-04	Facility Level Testi…	Client Initiated Testing…	Yes	Yes	Index Client Testing	Positive

An example of a wide table

PEPFAR TAP

How does MambaETL do this?

MambaETL main characteristics

- OpenMRS module
- Purely SQL compliant
- Highly configurable
- Schedulable

- It is packaged as an OpenMRS Module (OMOD), so it can be made part of a distribution
- The ETL process is based on stored procedures that can be scheduled
- Tables and columns to be flattened are highly configurable, giving flexibility to implementers
- The scripts can deployed in three different ways:
  - Same database (main OMRS database)
  - Separate database (analysis database) on the same server
  - Separate database in a different server (soon)

PEPFAR TAP

Very easy to customize

MambaETL automatically flattens all available encounters out of the box.

However an implementer might choose to customise how this is done through JSON configurations.

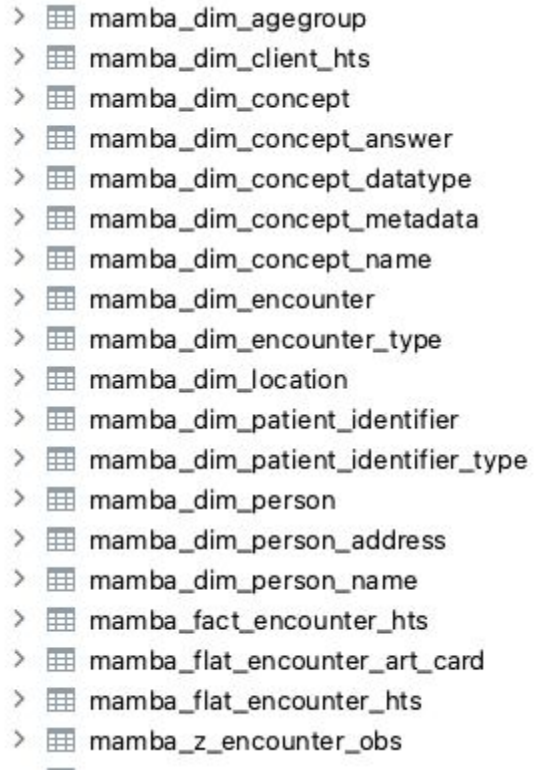
And that's all! MambaETL takes care of the rest


```
{
  "report_name": "HTS Report",
  "flat_table_name": "mamba_flat_encounter_hts",
  "encounter_type_uuid": "79c1f50f-f77d-42e2-ad2a-d29304dde2fe",
  "concepts_locale": "en",
  "table_columns": {
    "test_setting": "13abe5c9-6de2-4970-b348-36d352ee8eeb",
    "community_service_point": "74a3b695-30f7-403b-8f63-3f766461e104",
    "facility_service_point": "80bcc9c1-e328-47e8-affe-6d1bffe4adf1",
    "hts_approach": "9641ead9-8821-4898-b633-a8e96c0933cf",
    "pop_type": "166432AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA",
    "key_pop_type": "d3d4ae96-8c8a-43db-a9dc-dac951f5dcb3",
    "key_pop_migrant_worker": "63ea75cb-205f-4e7b-9ede-5f9b8a4dda9f",
    "key_pop_uniformed_forces": "b282bb08-62a7-42c2-9bea-8751c267d13e",
    "key_pop_transgender": "22b202fc-67de-4af9-8c88-46e22559d4b2",
    "key_pop_AGYW": "678f3144-302f-493e-ba22-7ec60a84732a",
    "key_pop_fisher_folk": "def00c73-f6d5-42fb-bcec-0b192b5be22d",
    "key_pop_prisoners": "8da9bf92-22f6-40be-b468-1ad08de7d457",
    "key_pop_refugees": "dc1058ea-4edd-4780-aeaa-a474f7f3a437",
    "key_pop_msm": "160578AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA",
    "key_pop_fsw": "160579AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA",
    "key_pop_truck_driver": "162198AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA",
    "key_pop_pwd": "365371fd-0106-4a53-abc4-575e3d65d372",
    "key_pop_pwid": "c038bff0-8e33-408c-b51f-7fb6448d2f6c",
    "sexually_active": "160109AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA",
    "unprotected_sex_last_12mo": "159218AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA",
    "sti_last_6mo": "156660AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA",
    "ever_tested_hiv": "1492AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA",
    "duration_since_last_test": "e7947a45-acff-49e1-ba1c-33e43a710e0d",
    "last_test_result": "159427AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA",
    "reason_for_test": "ce3816e7-082d-496b-890b-a2b169922c22",
    "pretest_counselling": "de32152d-93b0-412a-908a-20af0c46f215",
    "type_pretest_counselling": "0473ec07-2f34-4447-9c58-e35a1c491b6f",
```

A clean set of tables

The final product, the flat tables, can be easily imported and used in any reporting tool

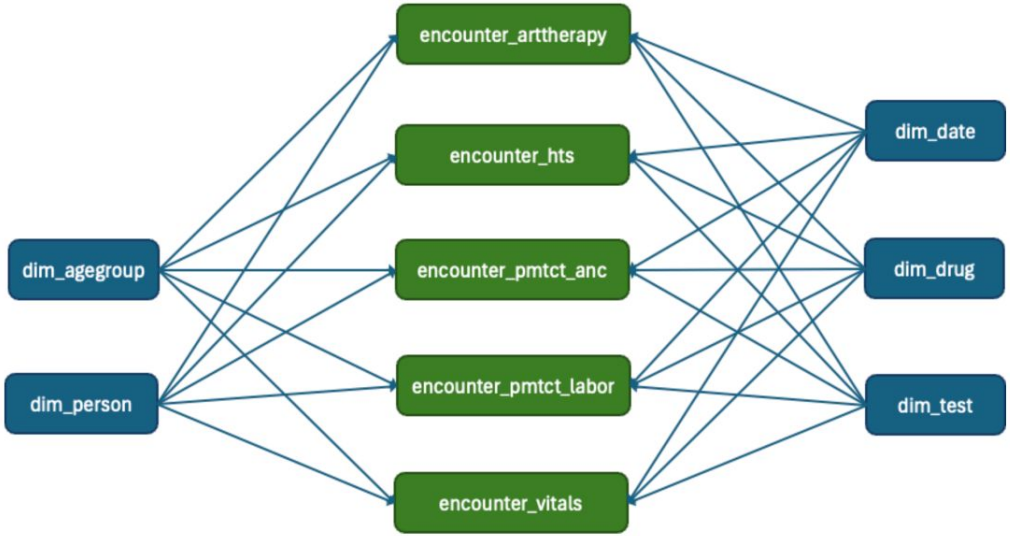
And because the data is persisted, reporting is fast



PEPFAR TAP

MambaETL: data model

Creates a simple **dimensional model** linking all encounters to a client, facility and date



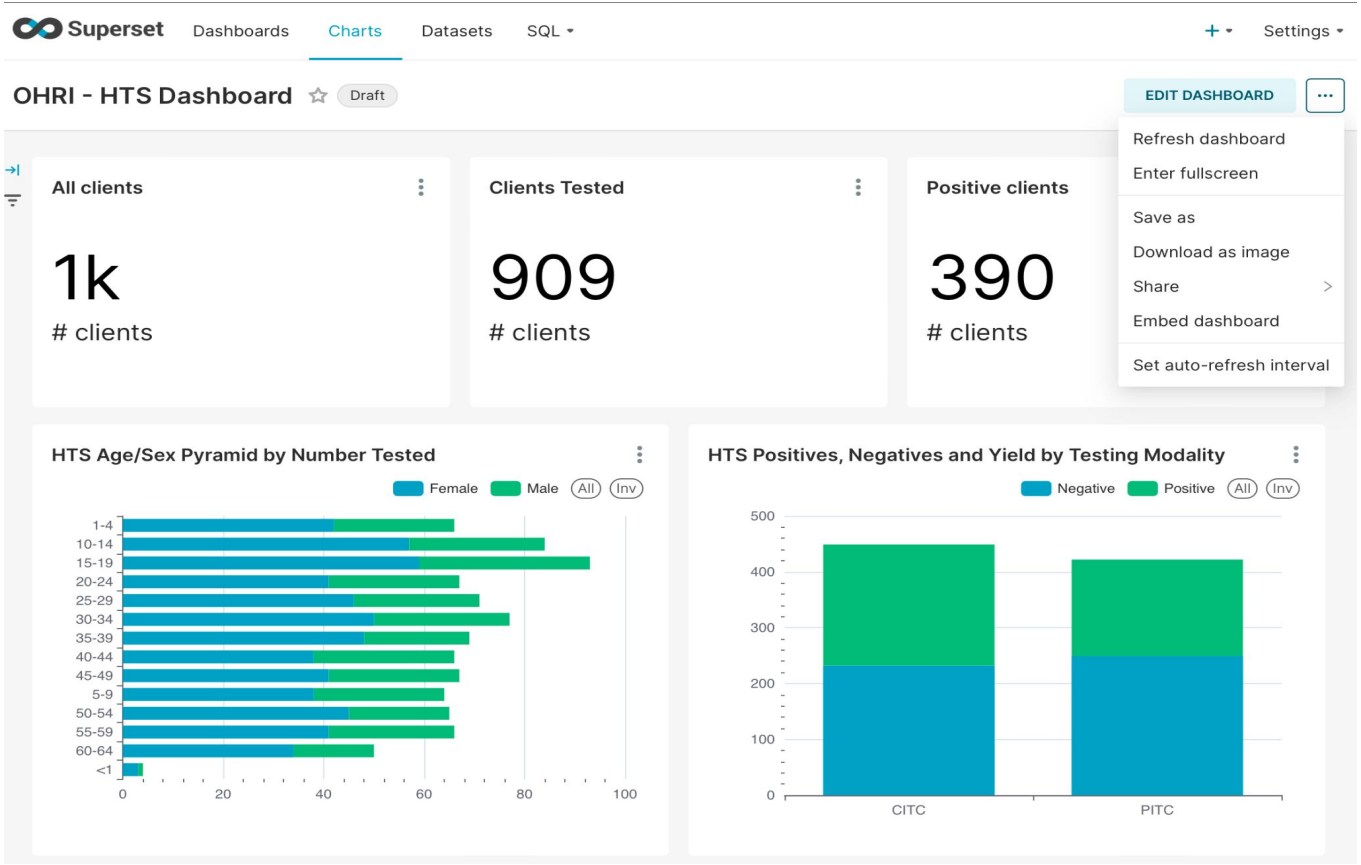
Once MambaETL has put all tables in place,

we can connect any type of reporting tool to it !
:-)

PEPFAR

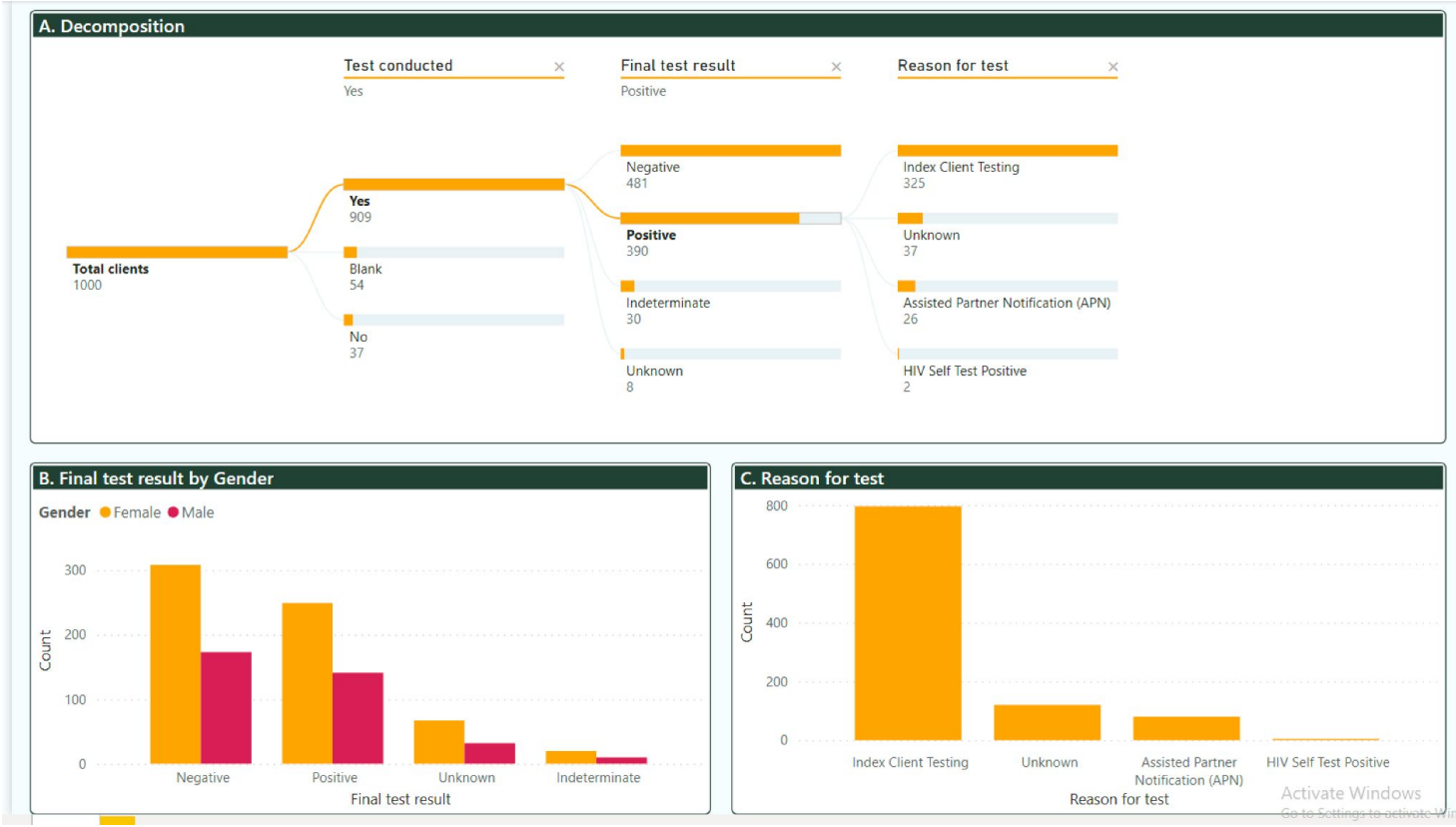
Technical Assistance Platform

Superset



PEPFAR Technical Assistance Platform

PowerBI



PEPFAR Technical Assistance Platform

Jasper Reports

DATIM MER 2.0 (Ver 2.6)

ORGANIZATION			
DATA SET		MER: FACILITY	
PERIOD	June		

DSD: HTS_TST (Facility)

Auto-Calculate: Number of individuals who received HIV Testing Services(HTS) and received their results

Numerator	Positive
89	47

Key Pop Type	Positive	Negative	Sub Totals
FSW	3	1	4
MSM	31	38	69
Prisoner	6	3	9
PWID	6	0	6
TRANS	1	0	1

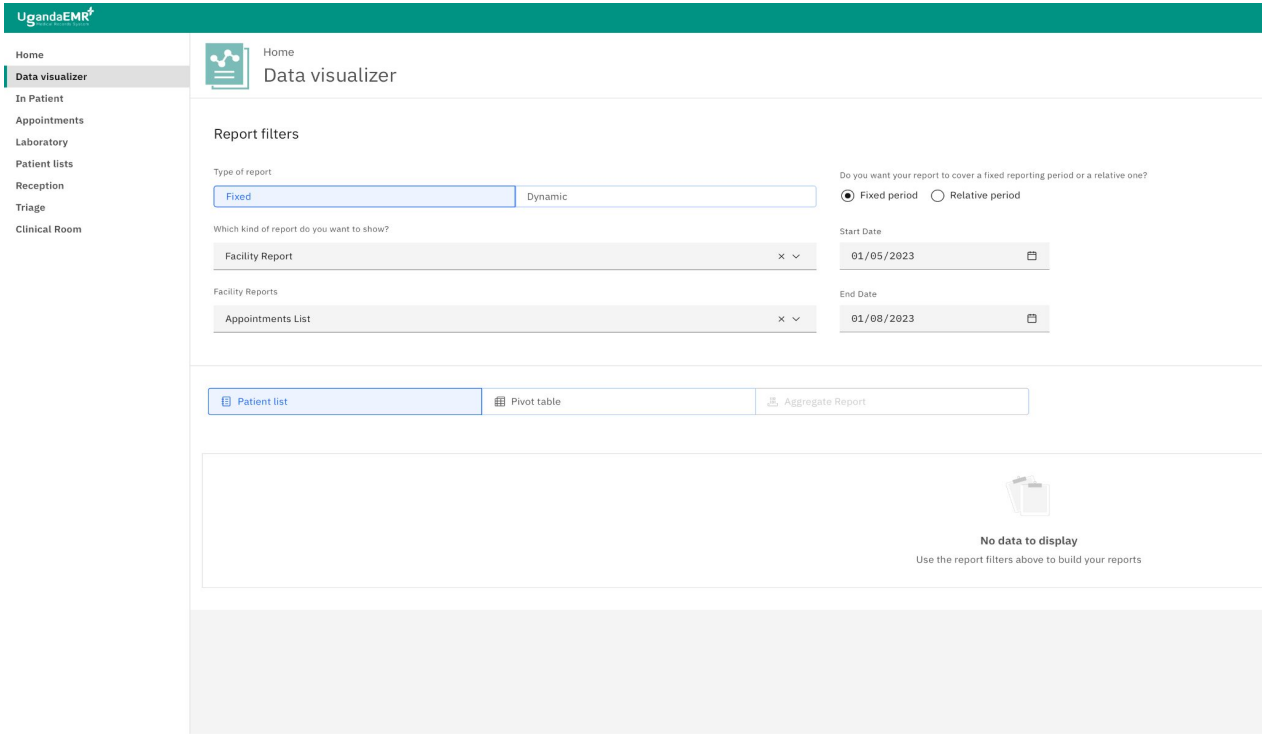
HTS_TST (Facility) – PITC Inpatient

		<1	1-4	5-9	10-14	15-19	20-24	25-29	30-34	35-39	40-44	45-49	50+	Sub Totals
Positive	Female	0	0	0	0	0	0	0	0	0	0	0	2	2
Positive	Male	0	1	1	1	0	0	0	1	0	0	0	1	5
Negative	Female	0	0	1	0	1	0	0	0	1	0	0	1	4
Negative	Male	0	0	2	0	0	0	0	0	0	0	0	2	4

PEPFAR Technical Assistance Platform

HTS MER Reports

UgandaEMR o3 Data-visualiser

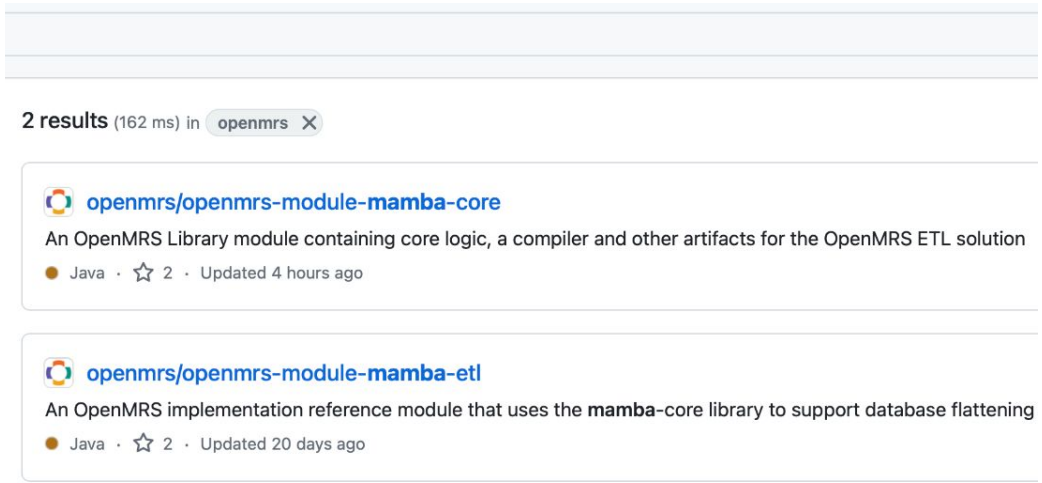


PEPFAR

Technical Assistance Platform

and...

- We have turned it into a generic, disease agnostic OpenMRS module
- Has become a community product
- Can be used by any OpenMRS installation, and **does not depend on O3**



Finally

PEPFAR

Technical Assistance Platform

Is MambaETL ready?

Yes!

We are in production in 3 different countries

- Rwanda MambaETL implementation
- Namibia PTracker implementation
- Uganda MambaETL implementation

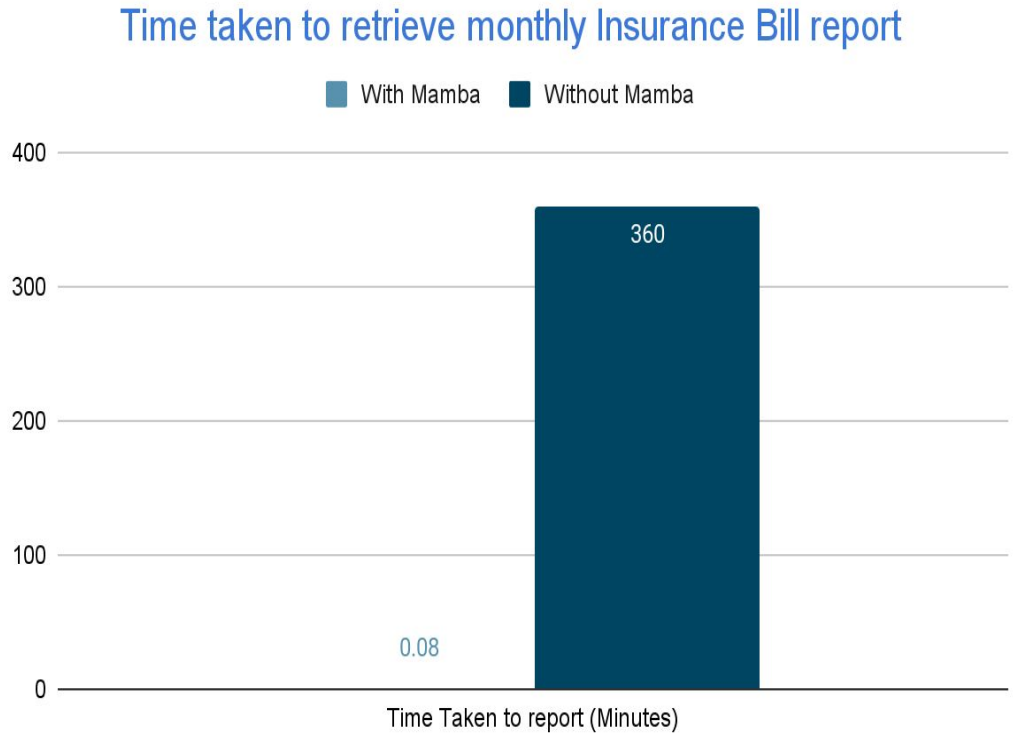
PEPFAR TAP

The Rwanda experience

Rwanda MoH faced performance issues reporting from OpenMRS data-model.

An insurance bill report for a month easily took more than 6 hours to fetch.

With MambaETL they could report on the same data in under 5 seconds



PEPFAR TAP

The UgandaEMR experience

performance challenges on large datasets

Limited capabilities in the Reporting module

Before MambaETL,

UgEMR used the OpenMRS Reporting module. It worked on smaller datasets (1-50 data elements) though with limitations.

On larger and complex reports (100-1000 data elements) reports took so long or simply hang.

PEPFAR TAP

UgandaEMR with MambaETL

A Case-study on 4 reports (before and after MambaETL)

Report name	Time taken to generate report (in minutes)	
	(with Reporting module)	(with MambaETL)
CQI HIV Audit tool Report	15	<3
HMIS 1061A	20	<2
TX CURR	5	< 1
TX NEW	5	< 1

UgandaEMR Data-visualiser

This is an O3 implementation data visualisation tool powered by the MambaETL.

Demo Link:
http://ugandaemr-demo.mets.or.ug/ugandaemr

MambaEL: On-going work

- Add support for other types of databases
- Deployment on a separate server, separate the transactional database from the reporting one
- Support for visual configurations
- Ethiopia OpenMRS implementation
- Community adoption and support

PEPFAR TAP

Thank you

MambaETL