Analytics Engine (including ETL and reporting improvement)

- Overview
  - Elevator Pitch
  - Background and strategic fit
- Goals & High-Level Requirements
  - Note: Feedback and input welcome!
  - MVP
  - MVP+
- Phases
  - Phase 1: A Working Proof of Concept ETL for a Large Site
    - Progress Update & Demo as of Oct 2020
  - Phase 2: Roll out to smaller sites
  - Phase 3: TBC - Other PEPFAR indicators? COVID indicators?
  - User Stories
  - Questions

Overview

Elevator Pitch

Why this squad? This squad’s work will shorten the time and improve the quality of using OMRS data in indicator reporting, reduce unplanned technical team member overtime, and make it easy to drill down to patient-level data to confirm the numbers are correct.

What do we do? We build and maintain the Analytics Engine for OpenMRS that runs in parallel to the EMR, which makes reporting much easier for Implementers, and will enable Clinical Decision Support in the not-too-distant future. All the work for integrating this engine with OMRS (both directions) and support for downstream data science tools should be addressed by this team as well.

Background and strategic fit

Right now, it’s difficult to do modern analytics on OpenMRS data. One of the key opportunities for greater collaboration within the OpenMRS community is around reporting through ETL (Extract, Transform, and Load). This was underscored during discussions at OMRS19. The manual process of designing and developing ETL queries is time-consuming, complex, and is being tackled one implementation at a time.

We have prototyped ETL approaches with both Spark and FHIR models.

We are handling most of the PEPFAR MER indicators using the traditional ETL pipeline, but we don't have a modular approach that allows more aggregate level analytics. Want to extract data out of OMRS in a structure that is easier to query so that it is easier to query specific indicators.

Goals & High-Level Requirements

Note: Feedback and input welcome!

The Analytics Engine Squad is still in the requirements-gathering process. If you have examples of pain points, requirements, and work-arounds your team is using in the field, we’d love to hear from you! Join us in our weekly Squad meetings, post on our community forum Talk (adding the label analytics-engine), post in our slack channel #etl, or reach out to our friendly squad technical leads: allan kimaina Bashir Sadjad; PM Grace Potma

MVP

- **Concrete Problems**: Pick concrete problems to solve in smaller pieces - regularly deliver value that sites can use. (Hence starting with the 10 indicators in Phase 1 below.)
- **Generalizable**: An output that is general, that most implementations can use. (Since inputs are so different by organization.)
- **FHIR-based**: Build on top of a FHIR-based datawarehouse (assumes people are comfortable working with FHIR schemas, and comfortable writing complex SQL queries. Need to validate)
- **Scaleable**: e.g. Not going to be a single mySQL approach, because this doesn’t scale and doesn’t address the issues Implementations have already run into, where mySQL might be fine when you have small, single sites; but when you want to do data analysis on all them, you need a data warehouse where a single mySQL approach won't work. So we don't want to be locked-in to mySQL.
- **Useable**: Whatever we do, users need to be able to run SQL on the solution since that’s what most M&E/Data Science to simply write SQL to generate a report
- **Patient-level Value**: Whatever metric is required, there should be support to drill down further, all the way down to 1 record (easily able to go down to pt records). E.g. “Aggregate the # of pts with VL Suppression.” Ppl see # and say “this is an under/over report!”
- **Exposable**: It should be easy to use the data from within OMRS (exposing the data in OMRS should be easy).
- **Clear Guidance**: Provide simple notebook with examples on how to use, query, specific transformations Don’t assume ppl know how to run spark cluster or connect data to mySQL DB
- **Handle simple deployment**: Runs on single node.
MVP+

- **Handle more complex deployments.** Runs on *multiple nodes*. Everything should just work regardless of how many VMs (single or 10’s) - does not require/assume cluster management. E.g. shouldn’t expose dataflow clusters to end users.
- **(TBC) De-identified:** For some sites, the data in their warehouses *must* be de-identified for compliance. For a few others this creates more contraints than they want. Toggling de-identification for a site’s DB or Access Management may be a features instead of an MVP requirements.

# Phases

## Phase 1: A Working Proof of Concept ETL for a Large Site

**Launch Goal:** Release Proof of Concept for Ampath to try out. Ampath has biggest DB & Allan’s expertise. As a result of this launch, we want to be able to say “with a DB this size, we can run these obs and metrics successfully so it should work for smaller sites”.

**Approach Overview:** The test list below contains 10 Indicators with simple, moderate, and complex indicator examples. They will act like canaries for our first ETL PoC because they’ll be proof-points for whether the pipeline approach had the expected outcome or not.

**PEPFAR MER Aggregates Identified for ETL PoC:**

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TB_ART</td>
<td>Proportion of HIV-positive new and relapsed TB cases on ART during TB treatment</td>
</tr>
<tr>
<td>TB_PREV</td>
<td>Proportion of ART patients who started on a standard course of TB Preventive Treatment (TPT) in the previous reporting period who completed therapy</td>
</tr>
<tr>
<td>TB_STAT</td>
<td>Percentage of new and relapse TB cases with documented HIV status</td>
</tr>
<tr>
<td>TX_CURR</td>
<td>Number of adults and children currently receiving antiretroviral therapy (ART)</td>
</tr>
<tr>
<td>TX_MLNT</td>
<td>Number of ART patients (who were on ART at the beginning of the quarterly reporting period) and then had no clinical contact since their last expected contact</td>
</tr>
<tr>
<td>TX_NEW</td>
<td>Number of adults and children newly enrolled on antiretroviral therapy (ART)</td>
</tr>
<tr>
<td>TX_PVL</td>
<td>Percentage of ART patients with a suppressed viral load (VL) result (&lt;1000 copies/ml) documented in the medical or laboratory records/laboratory information systems (LIS) within the past 12 months</td>
</tr>
<tr>
<td>TX_RTT</td>
<td>Number of ART patients with no clinical contact (or ARV drug pick-up) for greater than 28 days since their last expected contact who restarted ARVs within the reporting period</td>
</tr>
<tr>
<td>TX_TTB</td>
<td>Proportion of ART patients screened for TB in the semiannual reporting period who start TB treatment.</td>
</tr>
</tbody>
</table>

**Patient-Level Indicators Required to generate these aggregates**

1. Enrollment Date - Previous Definition
2. ARV First Regimen Start Date - Previous Definition
3. VL Count Result - Previous Definition
4. VL Count Date - Previous Definition
5. Current and Previous ARV Meds - Previous Definition
6. TB Treatment Start Date - Previous Definitions
7. TB Treatment End Date - Previous Definitions
8. TB Prevention Start Date - Previous Definitions
9. TB Prevention End Date - Previous Definitions
10. Supporting Variables: patient_id, encounter_id, prev_rtc_date, is_clinical_encounter, location_id

**Progress Update & Demo as of Oct 2020**

https://docs.google.com/presentation/d/1ZCa3VXO1OvNB442YqB91kS5zMTjDdjn7A1IApwaP-g/edit#slide=id.g8f74bf3fe7_0_48
Phase 2: Roll out to smaller sites
Phase 3: TBC - Other PEPFAR indicators? COVID indicators?

User Stories
  • ...

Questions
  • ...