Data Flattening using Bahmni Mart

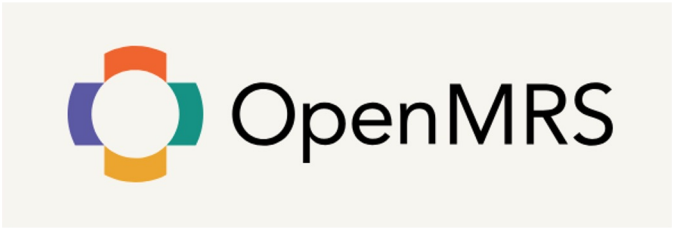
Pramida Tumma

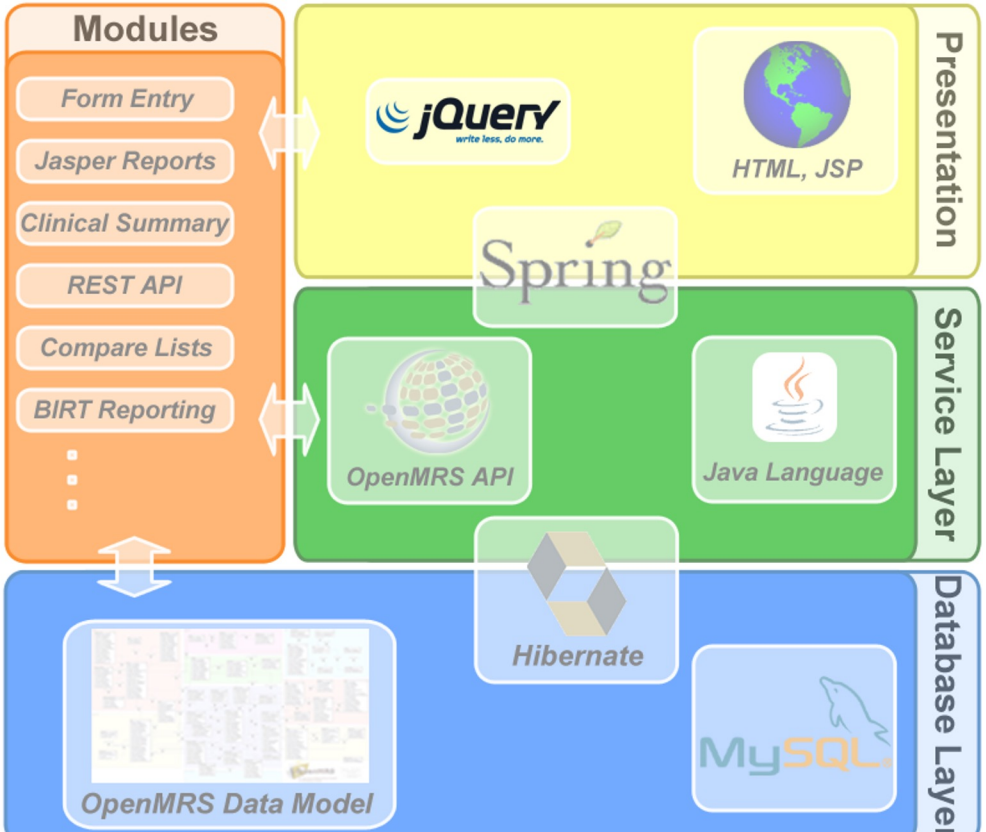
Agenda

- Introduction to Openmrs and Bahmni
- Problem and Solution provided
- Bahmni Mart technical details
- Analytical tool
- Drawbacks
- Impact of Bahmni Mart

OpenMRS

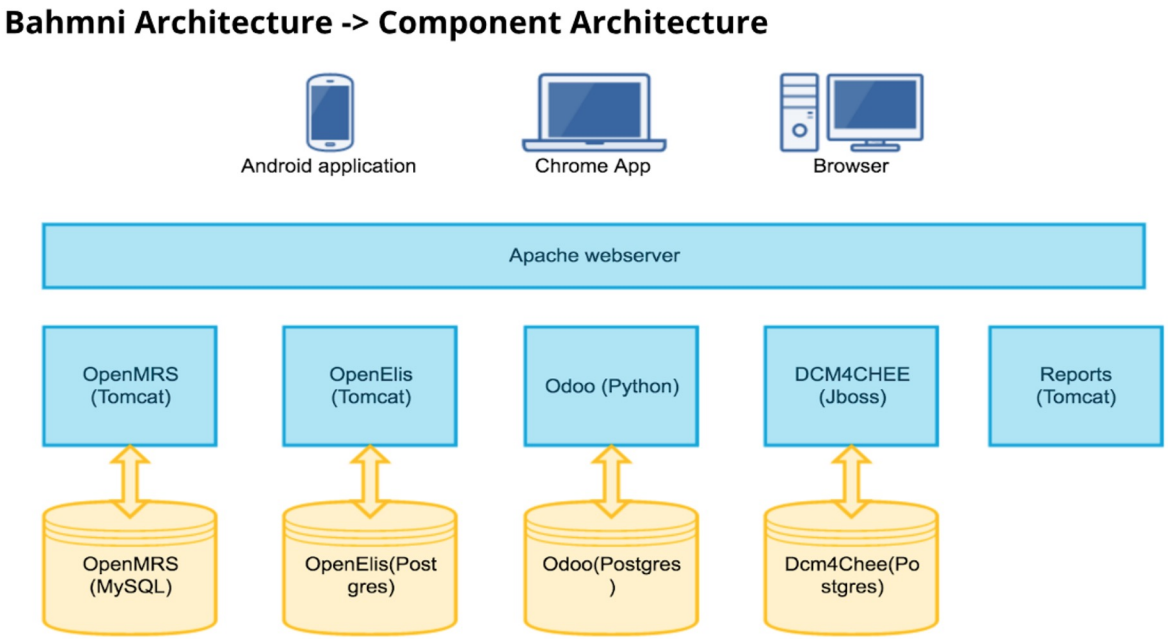
World's leading open source enterprise electronic medical record (EMR) system platform.





Bahmni

- Open Source EMR and Hospital System for low resource settings
- Distribution of OpenMRS, OpenELIS, OpenERP, DICOM and PACS
- Easy to Use



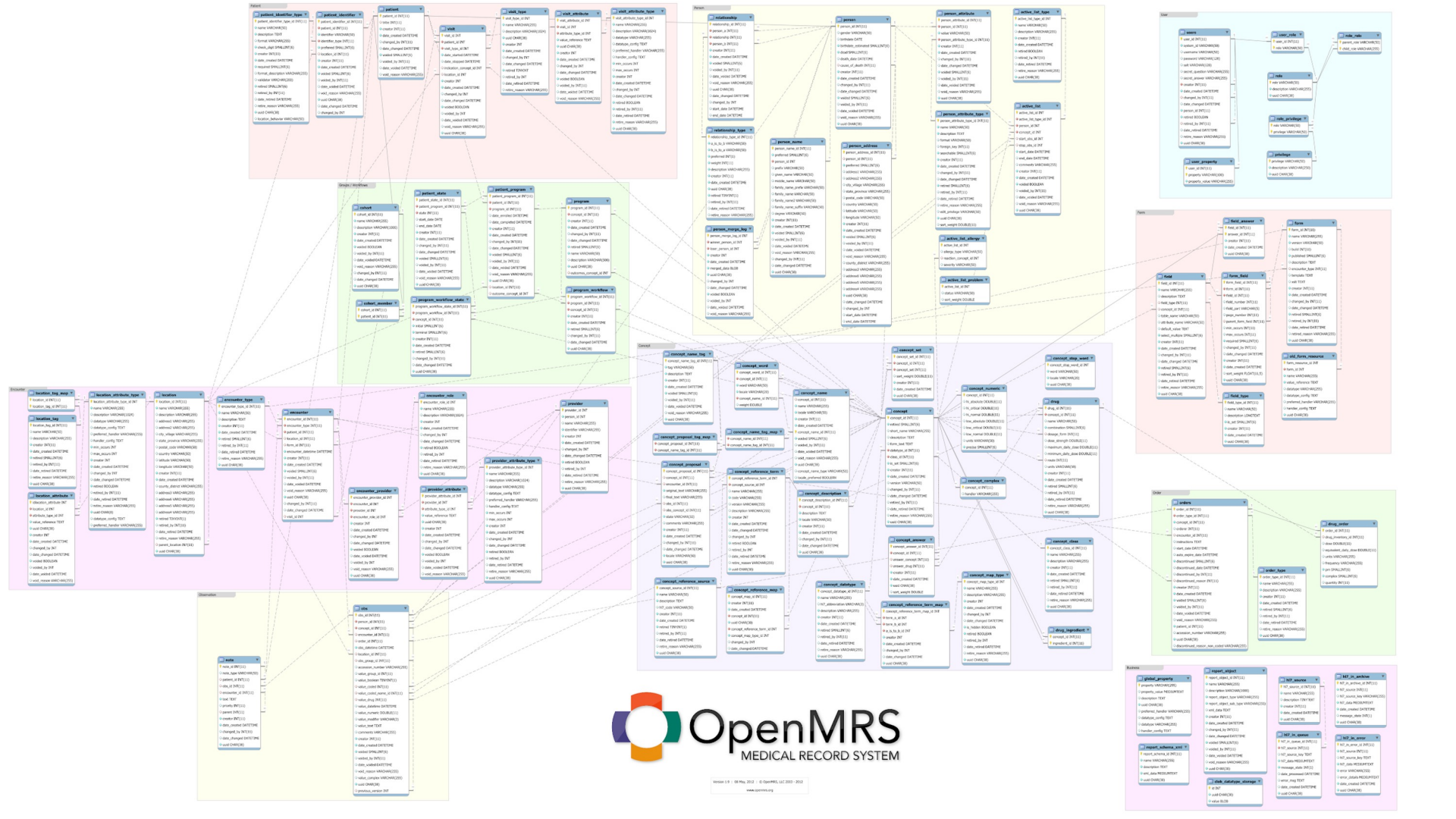
Indicators and Reports with Data

- Indicator report for every month
  - Time consuming
  - Need TW help for any new Indicator/Report

Indicator Report

Baseline Indicators	Jan-18	Feb-18	Mar-18	Apr-18	May-18	Jun-18	Jul-18	Aug-18	Sep-18	Oct-18	Nov-18	Dec-18
Files Received and entered in EMR	0	0	0	0	0	0	0	0	0	0	0	0
Validation Committee decision, new cases	0	0	0	0	0	0	0	0	0	0	0	0
Proportion of refused, all cases	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Proportion of postponed, all cases	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Patients arrived in Amman	35	42	60	51	47	43	58	41	52	55	30	11

- Adhoc requests like for how many patients was this drug prescribed in last 2 months
  - Trained people on field in SQL, OpenMRS data model


OpenMRS
MEDICAL RECORD SYSTEM

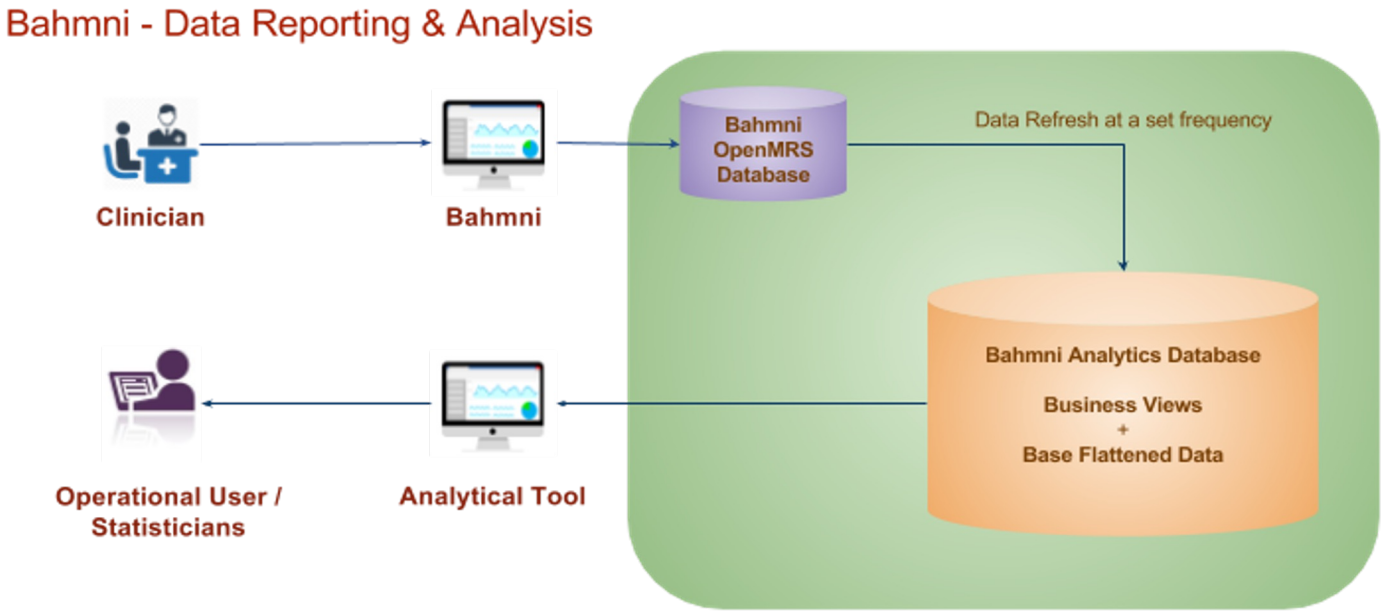
Bahmni Mart - Business Objective

- Build a **Bahmni Analytics Data Model** on the complex **hierarchical** openMRS data model that could be packaged with Bahmni or Bahmni Lite
- Will be a **Generic Flat Data Model** across implementations

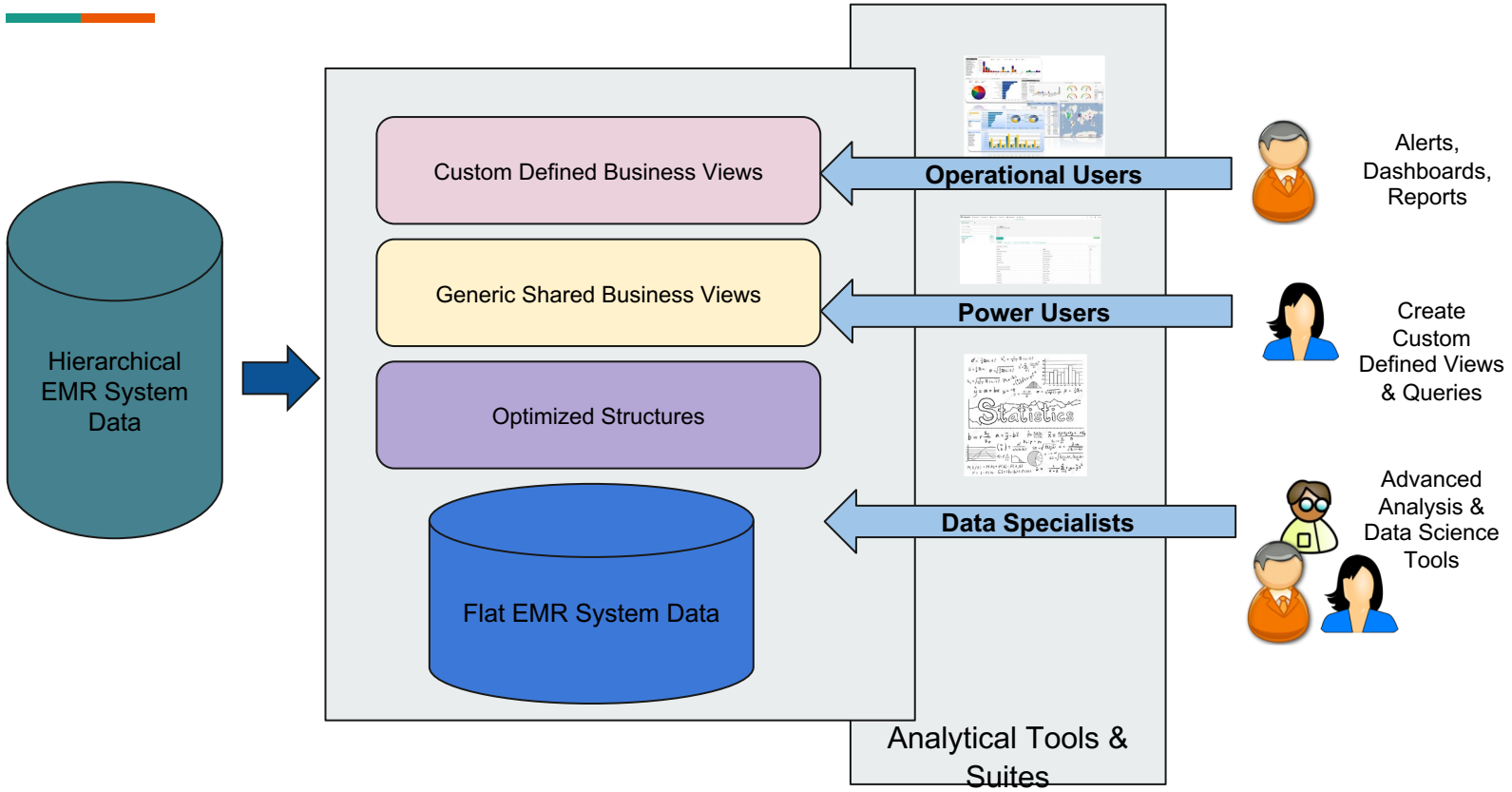
Value:

- Make it easier for various **Data Analytics tools** to directly **consume Bahmni Analytics database**

Reporting Platform



Analytics Information Architecture



Comparison View of Query Complexity

```sql
SELECT
  exitTypeConcept.name AS type_of_exit,
  count(person_id)     AS avg_patient_count
FROM obs exitTypeObs
LEFT OUTER JOIN concept_name exitTypeConcept
    ON exitTypeConcept.concept_id = exitTypeObs.value_coded AND
exitTypeConcept.voided IS FALSE
WHERE exitTypeObs.obs_id IN (
  SELECT obs_id
  FROM obs allObs
  WHERE allObs.concept_id = (
        SELECT concept_id
        FROM concept_name cn
        WHERE cn.name = "Type of Exit"
  )
)
AND exitTypeObs.voided IS FALSE
GROUP BY exitTypeObs.value_coded
ORDER BY count(person_id) DESC;
```

Add Comment

Collapse ↑

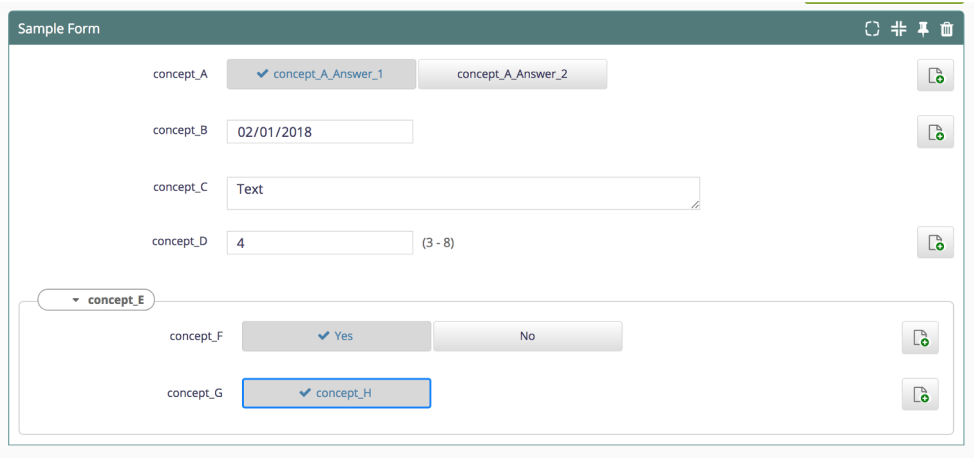
On Hierarchical OpenMRS based Database

On Bahmni Analytics DB

```sql
SELECT type_of_exit, count(*) as avg_patient_count
from exit
WHERE type_of_exit is not null
GROUP BY type_of_exit
ORDER BY count(*) DESC ;
```

Flattening Approach

Hierarchical data - Observations



Obs_id	Person_id	Concept_id	encounter_id	obs_group_id	value_coded	value_datetime	value_numeric	value_Text
1	100	3682 (concept id of "Sample Form")	49823					
2	100	3686 (concept id of "concept_C")	49823	1				Text
3	100	3683 (concept id of "concept_A")	49823	1	3684 (concept id of "concept_A_Answer_1")			
4	100	3688 (concept id of "concept_E")	49823	1				
5	100	3689 (concept id of "concept_F")	49823	4	1			
6	100	3690 (concept id of "concept_G")	49823	4	3691 (concept id of "concept_H")			
7	100	3687 (concept id of "concept_D")	49823	1			4	
8	100	3685 (concept id of "concept_B")	49823	1		2/1/18 0:00		

Output

id_sample_form	Person_id	encounter_id	concept_A	concept_B	concept_C	concept_D	concept_F	concept_G
1	100	49823	concept_A_Answer_1	2/1/18 0:00	Text	4	Yes	concept_H

Attribute tables for user defined fields - Person attributes

person_attribute_type_id	name	description	format
8	givenNameLocal	Name in Arabic	java.lang.String
9	familyNameLocal	Family Name in Arabic	java.lang.String
10	middleNameLocal	Middle Name in Arabic	java.lang.String
13	viber	Viber	java.lang.String
14	phoneNumber2	Phone Number 2	java.lang.String
15	facebook	Facebook	java.lang.String
16	whatsapp	WhatsApp	java.lang.String
17	emailAddress	Email Address	java.lang.String
18	phoneNumber1	Phone Number 1	java.lang.String
19	id3FullNameArabic	Full Name in Arabic	java.lang.String
20	id2DocNumber	ID Document Number	java.lang.String

person_attribute_id	person_id	value	person_attribute_type_id
1	124	124-givenNameLocal	8
2	124	124-familyNameLocal	9
3	124	2517446105	18
4	124	124-id1FullNameArabic	24
5	124	8940477532	31
6	124	142	35
7	124	124-id1FullNameEnglish	36
8	124	2018-07-26T00:00:00.000	39
9	124	220	41
10	124	124-occupation	43

Output:

person_id	campDistrict	campVillage	caretakerDob	caretakerGe	caretakerNa	caretakerNa	caretakerNat
124							
125			1963-01-10T	Male	125-caretake	125-caretake	Iraqi
126							
127							
128			1970-07-18T	Male	128-caretake	128-caretake	Iraqi
129			1976-11-22T	Male	129-caretake	129-caretake	Iraqi
130				Male		130-caretake	Iraqi
131			1972-05-04T	Male	131-caretake	131-caretake	Iraqi

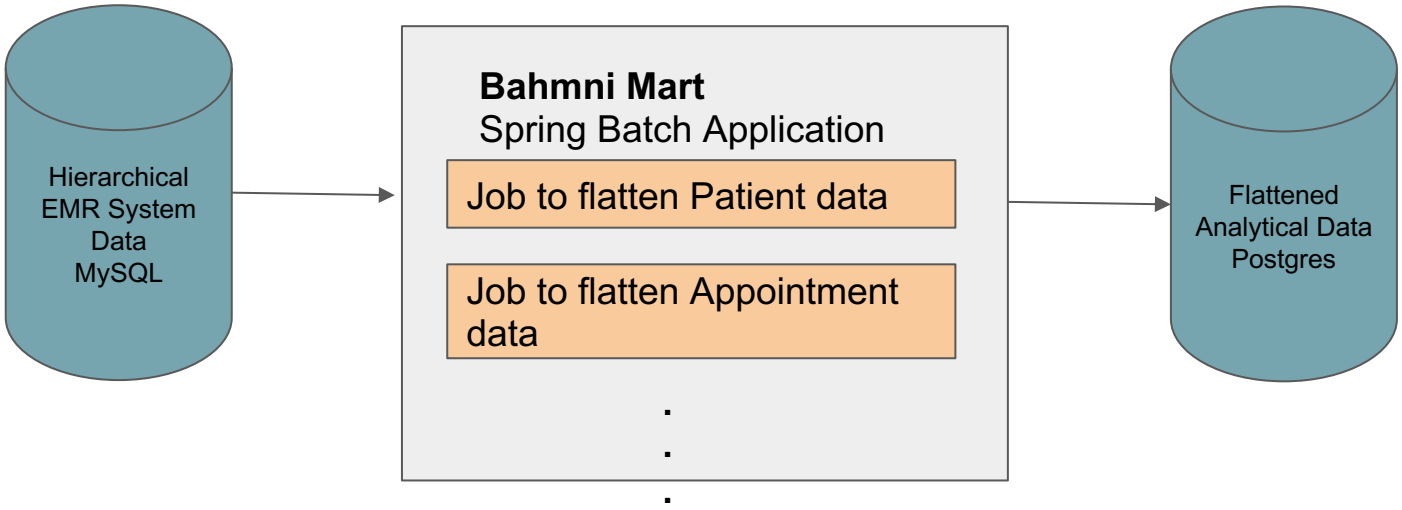
Data across multiple tables

Patient State - Patient, patient_program, program, patient_state, users, person_name, program_workflow_state, concept_view

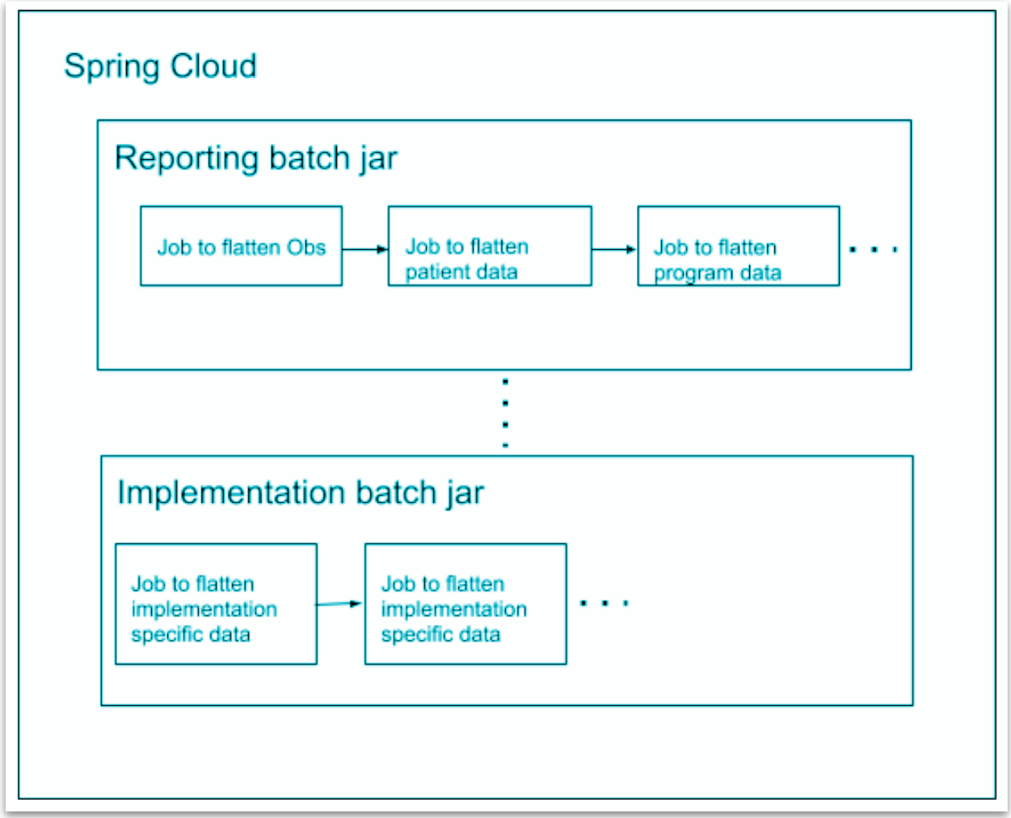
	patient_state_id	patient_program_id	patient_id	program_id	program_name	state	state_name	start_date	end_date	creator_id	
1	3588	1	124	1	Reconstructive Surgery	8	Network Follow-up	2015-05-22	<null>	22	M
2	2	2	125	1	Reconstructive Surgery	8	Network Follow-up	2013-08-01	2017-04-17	22	M
3	895	2	125	1	Reconstructive Surgery	10	Rehabilitation	2017-04-17	2017-05-30	22	M
4	1689	2	125	1	Reconstructive Surgery	8	Network Follow-up	2017-05-30	<null>	17	M
5	3	3	126	1	Reconstructive Surgery	8	Network Follow-up	2014-09-25	<null>	17	M
6	5	5	127	1	Reconstructive Surgery	2	Identification	2016-05-21	<null>	17	M
7	6	6	128	1	Reconstructive Surgery	8	Network Follow-up	2016-08-16	2017-04-02	17	M
8	1067	6	128	1	Reconstructive Surgery	10	Rehabilitation	2017-04-02	2017-08-16	22	M
9	2222	6	128	1	Reconstructive Surgery	8	Network Follow-up	2017-08-16	<null>	17	M
10	7	7	129	1	Reconstructive Surgery	8	Network Follow-up	2016-08-30	<null>	22	M
11	8	8	130	1	Reconstructive Surgery	2	Identification	2016-10-30	2017-05-22	22	M

Technical Solution

- ➔ Use Spring batch application to flatten the openmrs database
- ➔ Input database is Openmrs database (MySql)
- ➔ Output database is Mart database (Postgres)



Technical Solution - Extendable



Incremental Updates

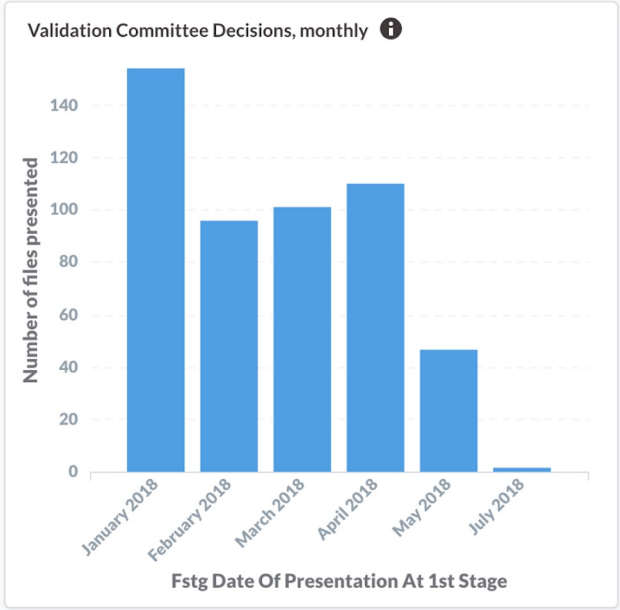
- ~2 GB - Half hour to flatten
- Using Event tables and Marker tables
- The tables used should be specified in the configuration
- With Incremental updates it takes up to 10 mins for a month long of data

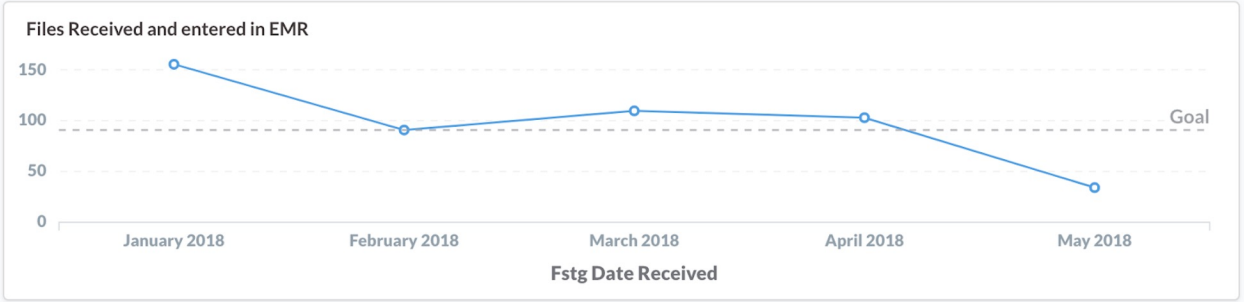
Other Highlights

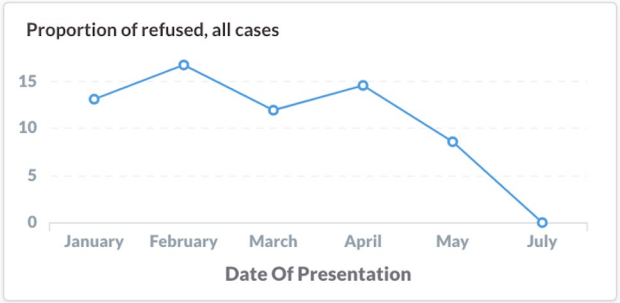
- Can be used by any Openmrs implementation
- Config driven jobs
- Can create a job using customSqls
- Exclusion of **PII data**
- Ignore columns that are not required
- Can have coded output using concept reference source maps
- Can create required views using config
- Ignore free text concepts as they are not used in analysis

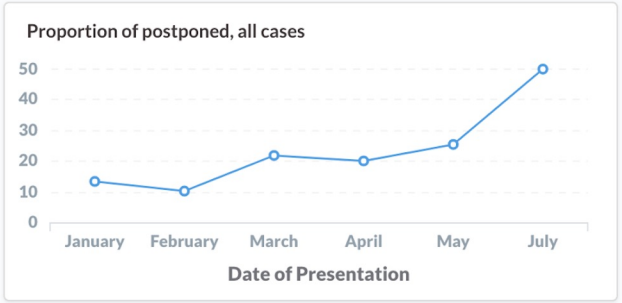
Analytical Tool

- Able to connect Mart database to any analytical tool like Tableu, PowerBI
- Open Source tool given by default is Metabase
- Took less than a week to build all the indicators in Metabase









Drawbacks

- Very Data specific
- Scalable?
- Small Data (Not BIG Data)

Impact of Bahmni Mart

- Able to build a tool on Bahmni Mart -> Bahmni - DHIS Integration
- Cost of generating reports and indicators in Bahmni got reduced to more than half
- People in the field are able to generate their own reports
- MSF is planning to purchase analytical tools like PowerBI, Tableu to explore more

Sample Config

```
{
  "name": "Obs Data",
  "type": "obs",
  "incrementalUpdateConfig": {
    "updateOn": "encounter_id",
    "eventCategory": "Encounter",
    "openmrsTableName": "encounter"
  },
  "separateTableConfig": {
    "enableForAddMoreAndMultiSelect": true,
    "separateTables": [
    ]
  },
  "conceptReferenceSource": "",
  "ignoreAllFreeTextConcepts": true,
  "columnsToIgnore": [
    "Image"
  ]
},
```

Thank you