Using open source to bring clarity to cancer

Tanvi Pal (Sr Software Engineer)
Stephen Jakubowicz (Product Manager)

COTA

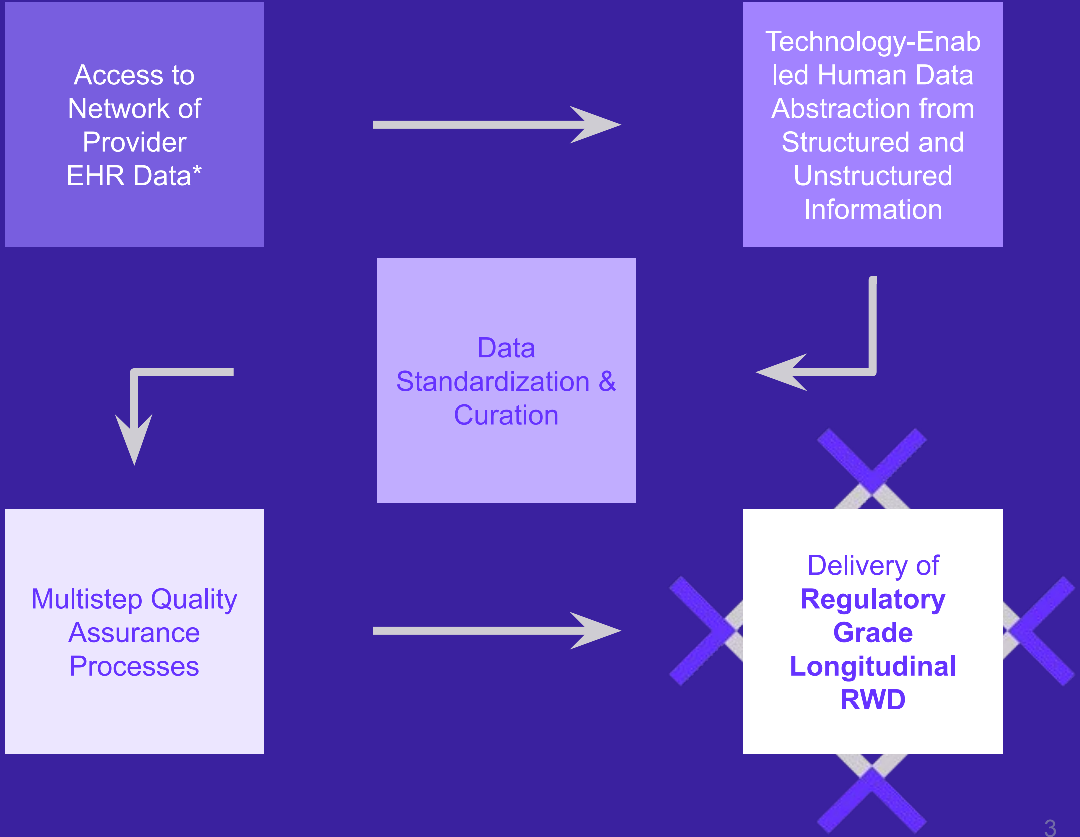
Agenda

- COTA overview and RWA introduction
- How open source has helped us?
- RWA tech stack
- Angular+Plotly.js example
- Cube.js
    - Pre-aggregations example
    - Boolean logic example
    - Asynchronous example

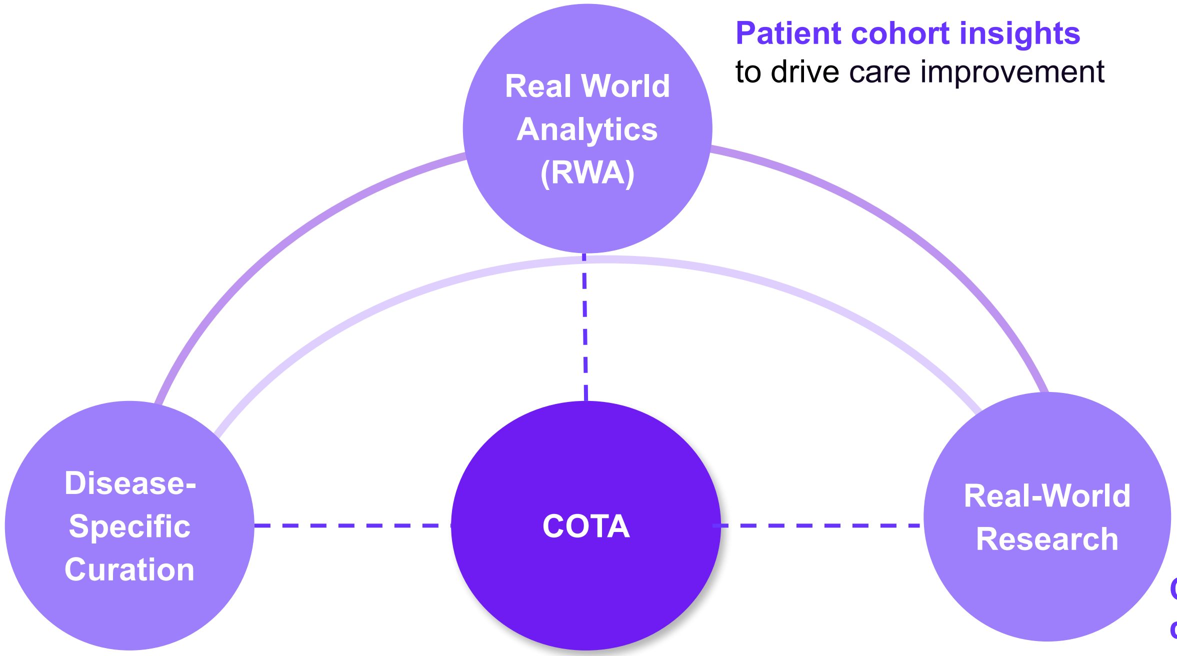
COTA Overview

COTA is a healthcare technology and services company founded in 2011 by doctors, engineers, and data scientists to create clarity from fragmented and often-inaccessible real-world data.

By using our proprietary technology, advanced analytics, and deep expertise to organize complex data, we provide a comprehensive picture of cancer that can be used to advance patient care and research.


Access to Network of Provider EHR Data*
Technology-Enabled Human Data Abstraction from Structured and Unstructured Information
Data Standardization & Curation
Multistep Quality Assurance Processes
Delivery of **Regulatory Grade Longitudinal RWD**

Powered by our platform, COTA offers providers a suite of solutions for not only research, but also to measure and track care delivery and quality



Patient cohort insights to drive care improvement

Disease-specific comprehensive curation powering care and research insights

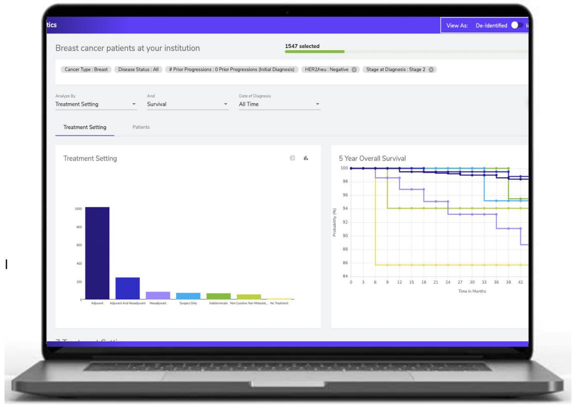
COTA's collective de-identified dataset to power real-world data research

COTA's RWA is a web application that provides the following core tools to answer key questions for hospitals

Provider Systems

1. Normalization, classification and enrichment

COTA's RWA



Breast cancer patients at your institution

1547 selected

Cancer Type : Breast | Disease Status : All | # Prior Progressions : 0 Prior Progressions (Initial Diagnosis) | HER2/neu : Negative | Stage at Diagnosis : Stage 2

Analyze By: Treatment Setting
And: Survival
Date of Diagnosis: All Time

2. Query and explore

3. Compare treatments and associated outcomes

4. Track key operational metrics and clinical insights

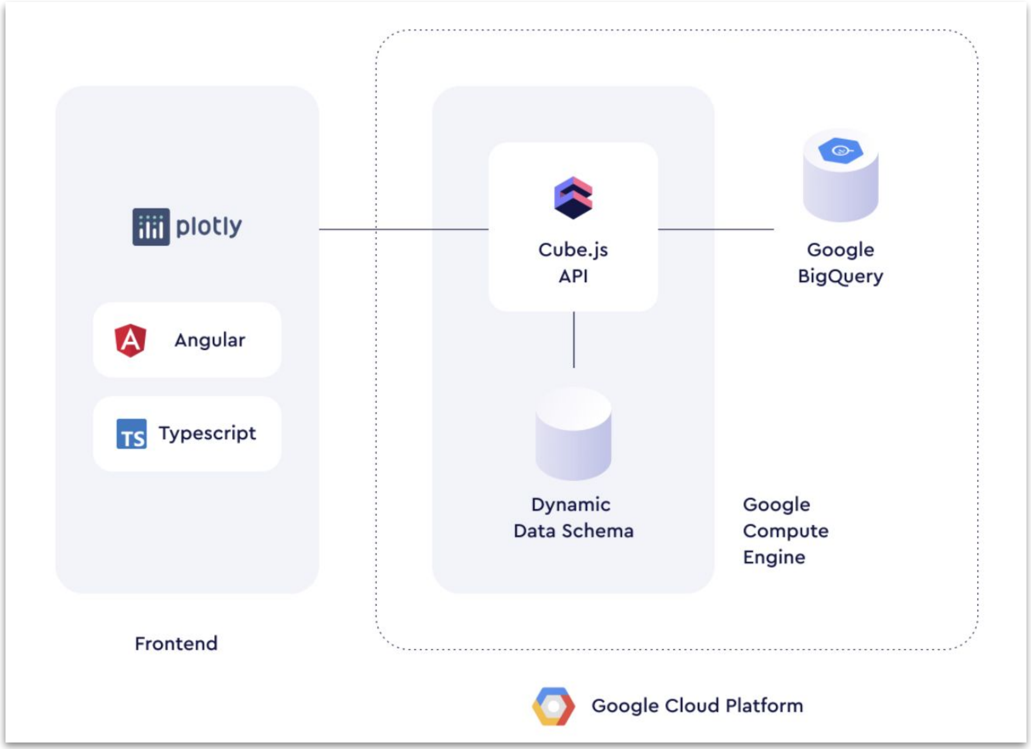
5. Longitudinal patient timeline and dashboard

How has open source helped us?

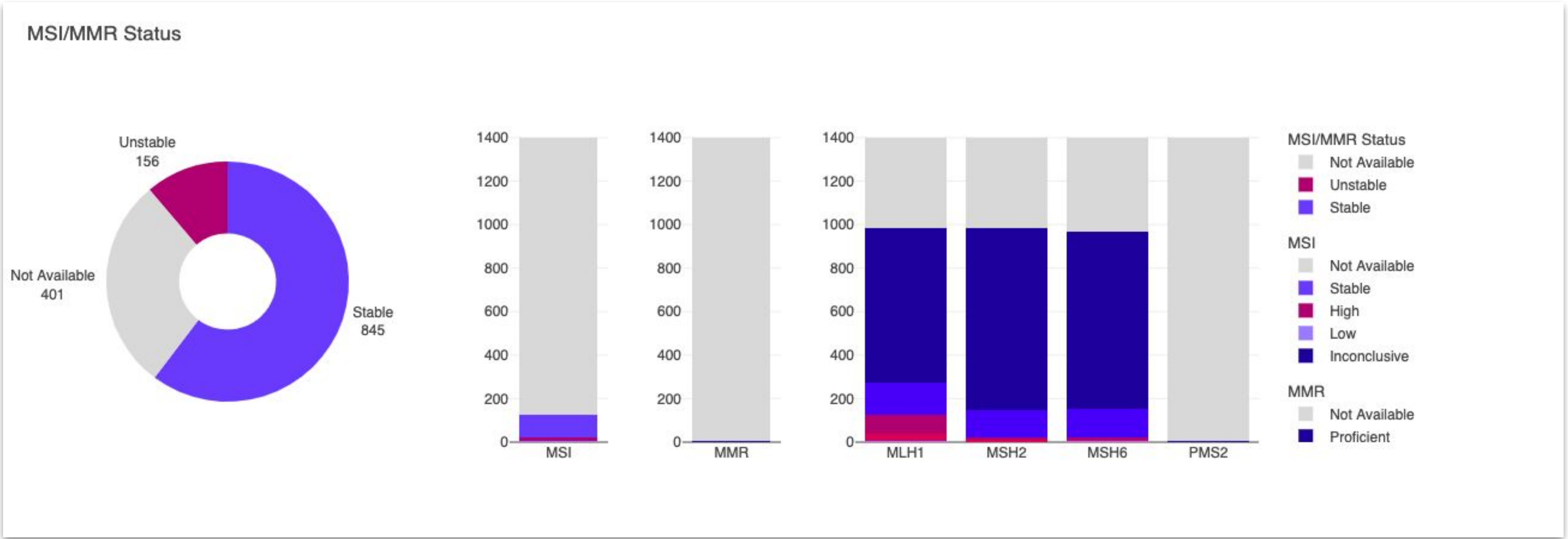
- Moved from using proprietary product to open source, we realised the following advantages :
  - Cost effective
  - Faster development
  - Customization possible
  - No vendor lock-ins, so update or replace if not supporting your cause
  - Better support and quicker iterations

RWA Tech Stack

- Angular
- Plotly
- Cube.js
- Codecept
- Node.js
- TypeScript
- Libraries
  - Moment
  - Lodash
- And many more



Angular / Plotly.js - Subplots

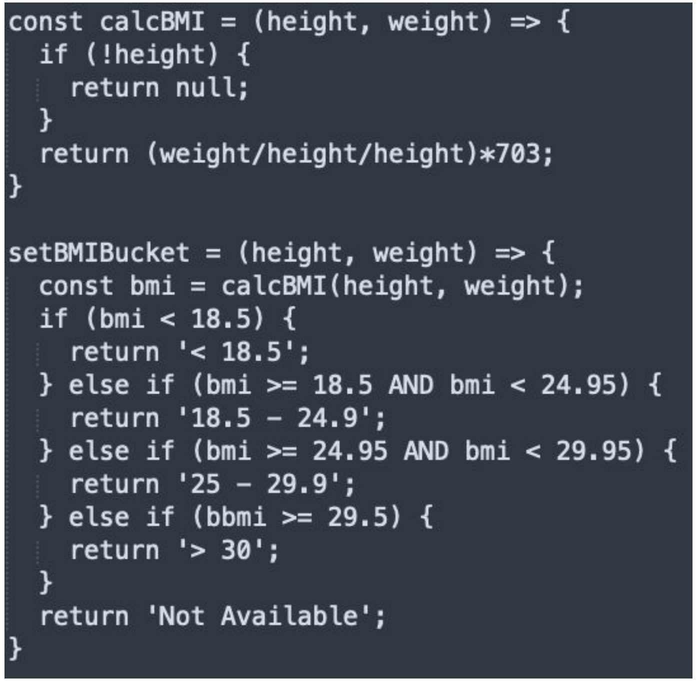


Cube.js - Flexibility, support, community

- COTA RWA was seeded along same time as Cube.js was setting up its foot in the OS world. COTA RWA and Cube.js has grown up together!
- Community effect was demonstrated since early stage. With limited capability at start, both the products influenced each-other and benefit from it.
- Contributions were done from product perspective and issues / nice-to-have-features were reported.
- Cube.js community is growing. Support from and within the community is awesome!

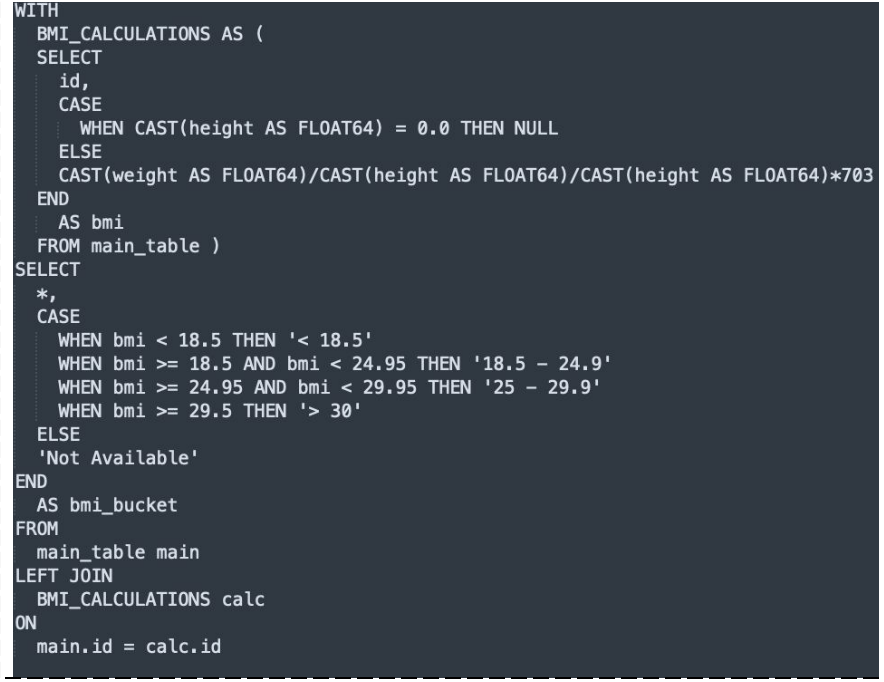
Cube.js - Pre-Agg

Data Massaging in Service Layer



```
const calcBMI = (height, weight) => {
  if (!height) {
    return null;
  }
  return (weight/height/height)*703;
}

setBMIBucket = (height, weight) => {
  const bmi = calcBMI(height, weight);
  if (bmi < 18.5) {
    return '< 18.5';
  } else if (bmi >= 18.5 AND bmi < 24.95) {
    return '18.5 - 24.9';
  } else if (bmi >= 24.95 AND bmi < 29.95) {
    return '25 - 29.9';
  } else if (bbmi >= 29.5) {
    return '> 30';
  }
  return 'Not Available';
}
```

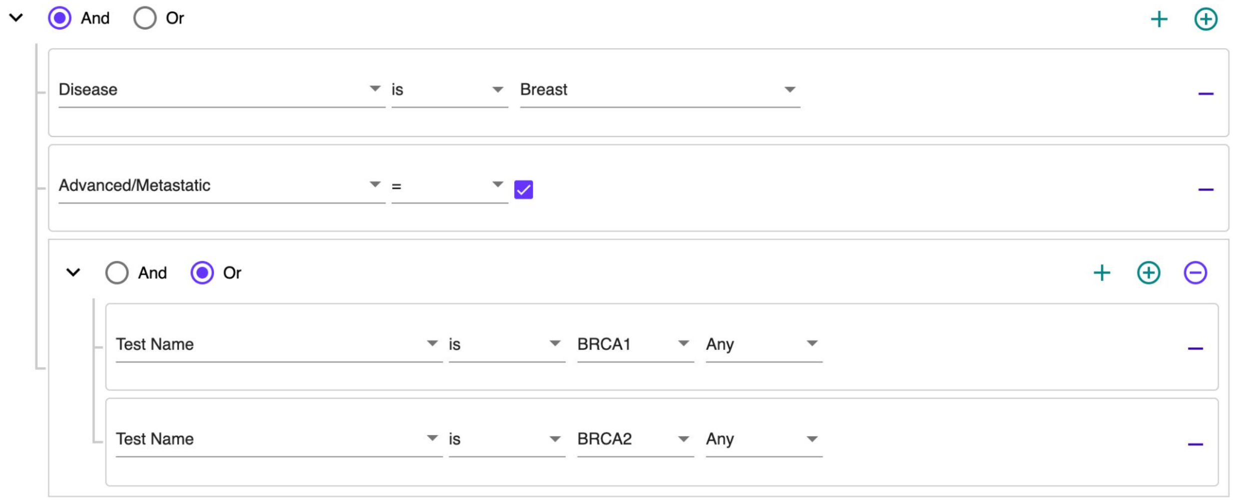
Pre Agg in Data Layer



```
WITH
  BMI_CALCULATIONS AS (
  SELECT
    id,
    CASE
      WHEN CAST(height AS FLOAT64) = 0.0 THEN NULL
    ELSE
    CAST(weight AS FLOAT64)/CAST(height AS FLOAT64)/CAST(height AS FLOAT64)*703
  END
    AS bmi
  FROM main_table )
SELECT
  *,
  CASE
    WHEN bmi < 18.5 THEN '< 18.5'
    WHEN bmi >= 18.5 AND bmi < 24.95 THEN '18.5 - 24.9'
    WHEN bmi >= 24.95 AND bmi < 29.95 THEN '25 - 29.9'
    WHEN bmi >= 29.5 THEN '> 30'
  ELSE
  'Not Available'
END
  AS bmi_bucket
FROM
  main_table main
LEFT JOIN
  BMI_CALCULATIONS calc
ON
  main.id = calc.id
```

Cube.js - Boolean Logic

Metastatic breast cancer patient, should have minimum 1 BRCA (a specific marker) test



[✓ And] Or

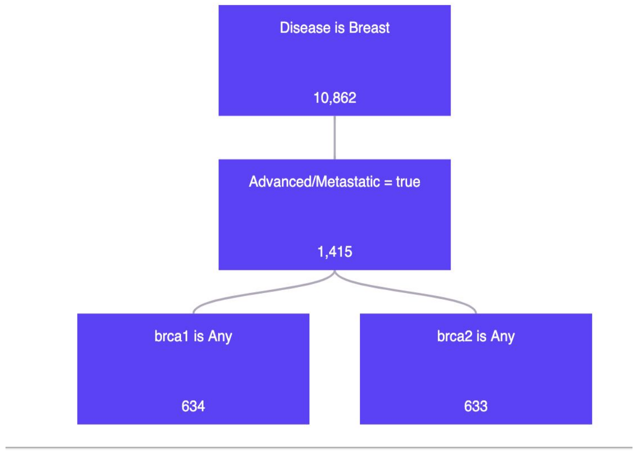
Disease is Breast

Advanced/Metastatic = ✓

And [✓ Or]

Test Name is BRCA1 Any

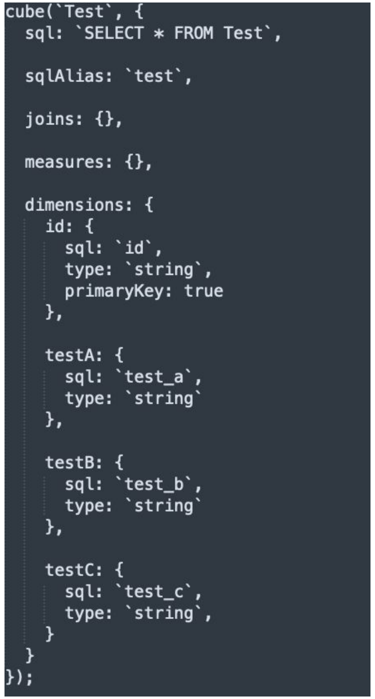
Test Name is BRCA2 Any



Total Records 640

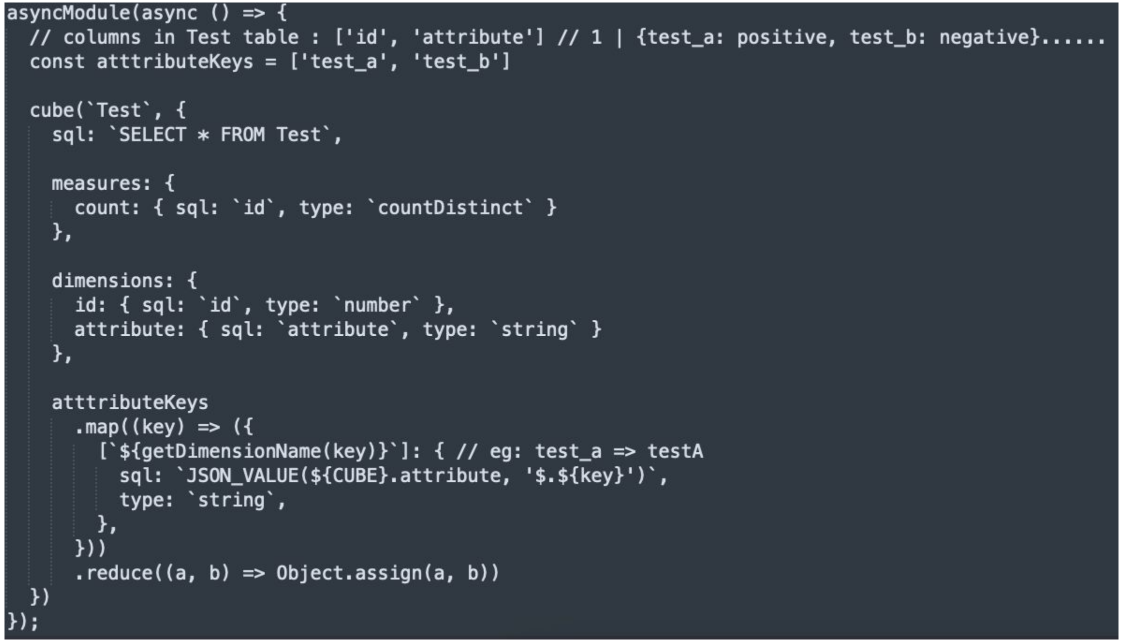
Cube.js - Async

Static Schema



```
cube(`Test`, {
  sql: `SELECT * FROM Test`,

  sqlAlias: `test`,

  joins: {},

  measures: {},

  dimensions: {
    id: {
      sql: `id`,
      type: `string`,
      primaryKey: true
    },

    testA: {
      sql: `test_a`,
      type: `string`
    },

    testB: {
      sql: `test_b`,
      type: `string`
    },

    testC: {
      sql: `test_c`,
      type: `string`,
    }
  }
});
```

Dynamic Schema



```
asyncModule(async () => {
  // columns in Test table : ['id', 'attribute'] // 1 | {test_a: positive, test_b: negative}......
  const atttributeKeys = ['test_a', 'test_b']

  cube(`Test`, {
    sql: `SELECT * FROM Test`,

    measures: {
      count: { sql: `id`, type: `countDistinct` }
    },

    dimensions: {
      id: { sql: `id`, type: `number` },
      attribute: { sql: `attribute`, type: `string` }
    },

    atttributeKeys
      .map((key) => ({
        [`${getDimensionName(key)}`]: { // eg: test_a => testA
          sql: `JSON_VALUE(${CUBE}.attribute, '$.${key}')`,
          type: `string`,
        },
      }))
      .reduce((a, b) => Object.assign(a, b))
  })
});
```

Thank you!