

Our journey through the pipeline

This presentation will explore our approach to building a dashboard API from several dynamic database sources

Jack Dench, Igor Langoni, Gina Medina, Hilary Smith and Niki Stefanopoulou



Agenda

1. EDA
2. Goal
3. Solution strategy
4. ETL
5. Flask API
6. Results
7. Areas for improvement

Exploratory Data Analysis (EDA)

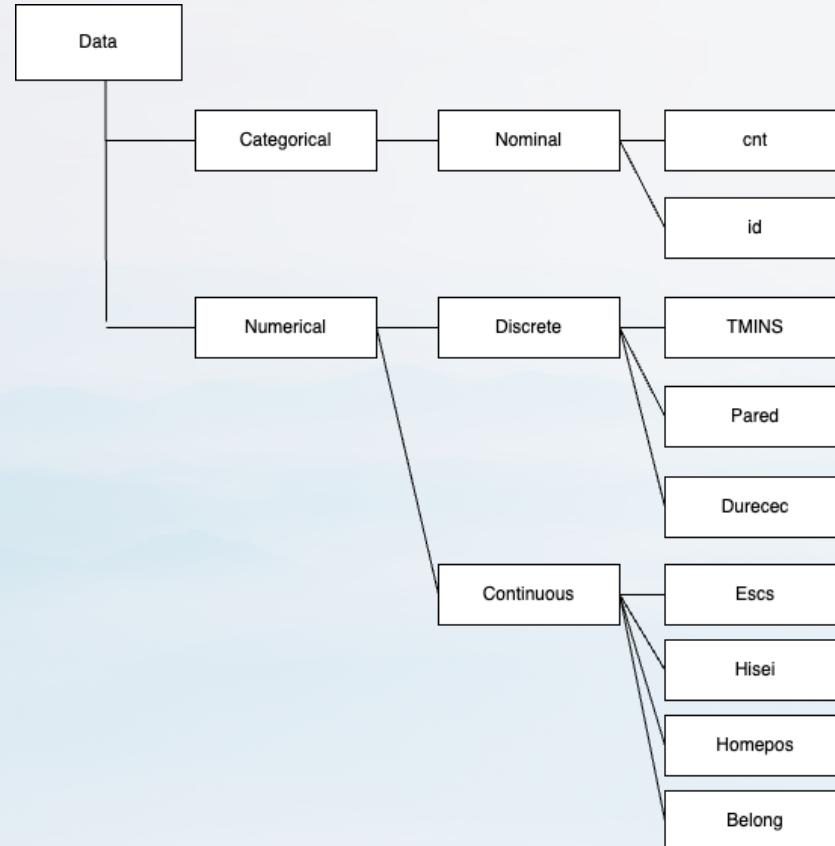
Total submissions

Submissions over time

Learning hours per week

Escs

Early education and Belonging



Goal



CHART
Early Education And Sense Of Belonging
A bubble plot of years in early education vs. self-reported sense of belonging, with the bubble representing the number of country submissions.
[Click here for the full component specification.](#)

Polling URL (will be polled once per second)
 Save

CHART
Number Of Submissions
A running count of all submissions across all countries.
[Click here for the full component specification.](#)

Polling URL (will be polled once per second)
 Save



CHART
Submissions Over Time
A moving line chart of submissions over time, by hour.
[Click here for the full component specification.](#)

Polling URL (will be polled once per second)
 Save

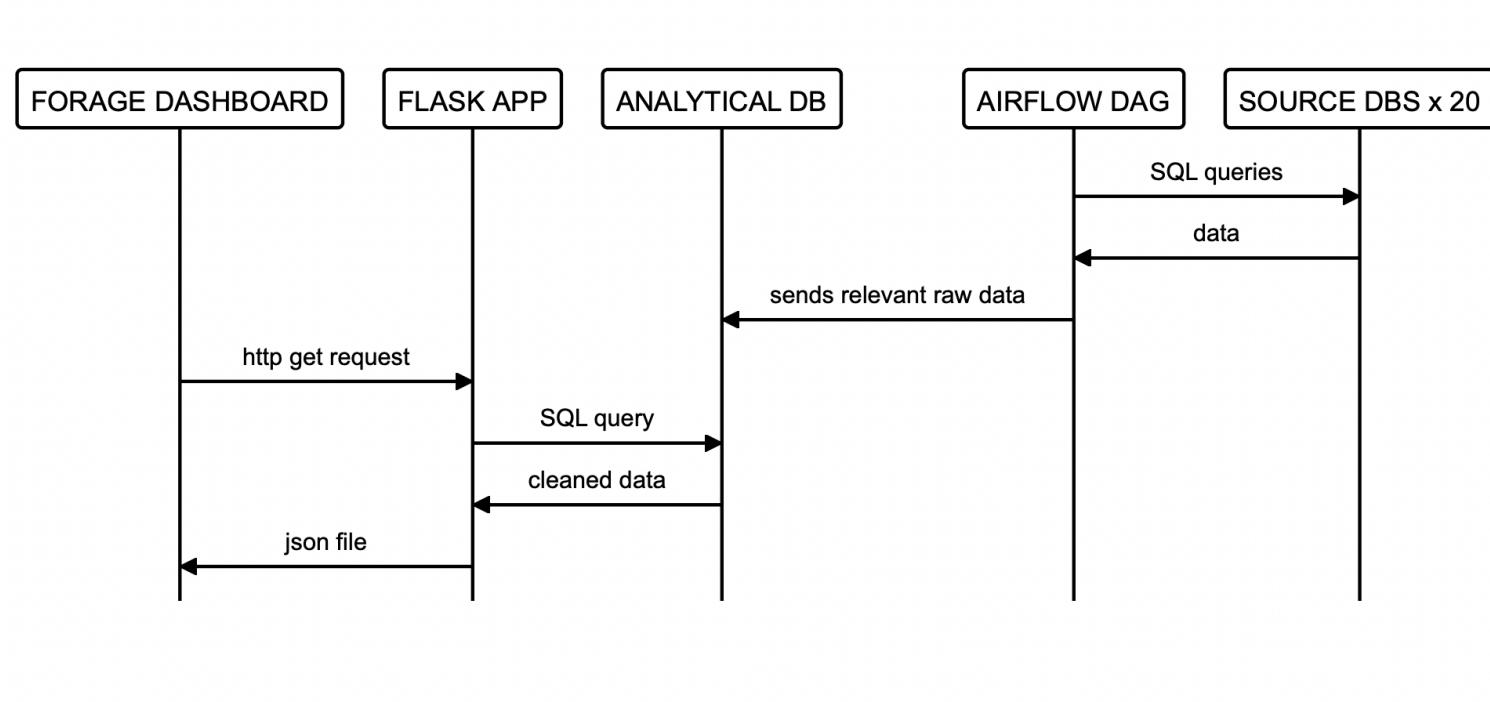


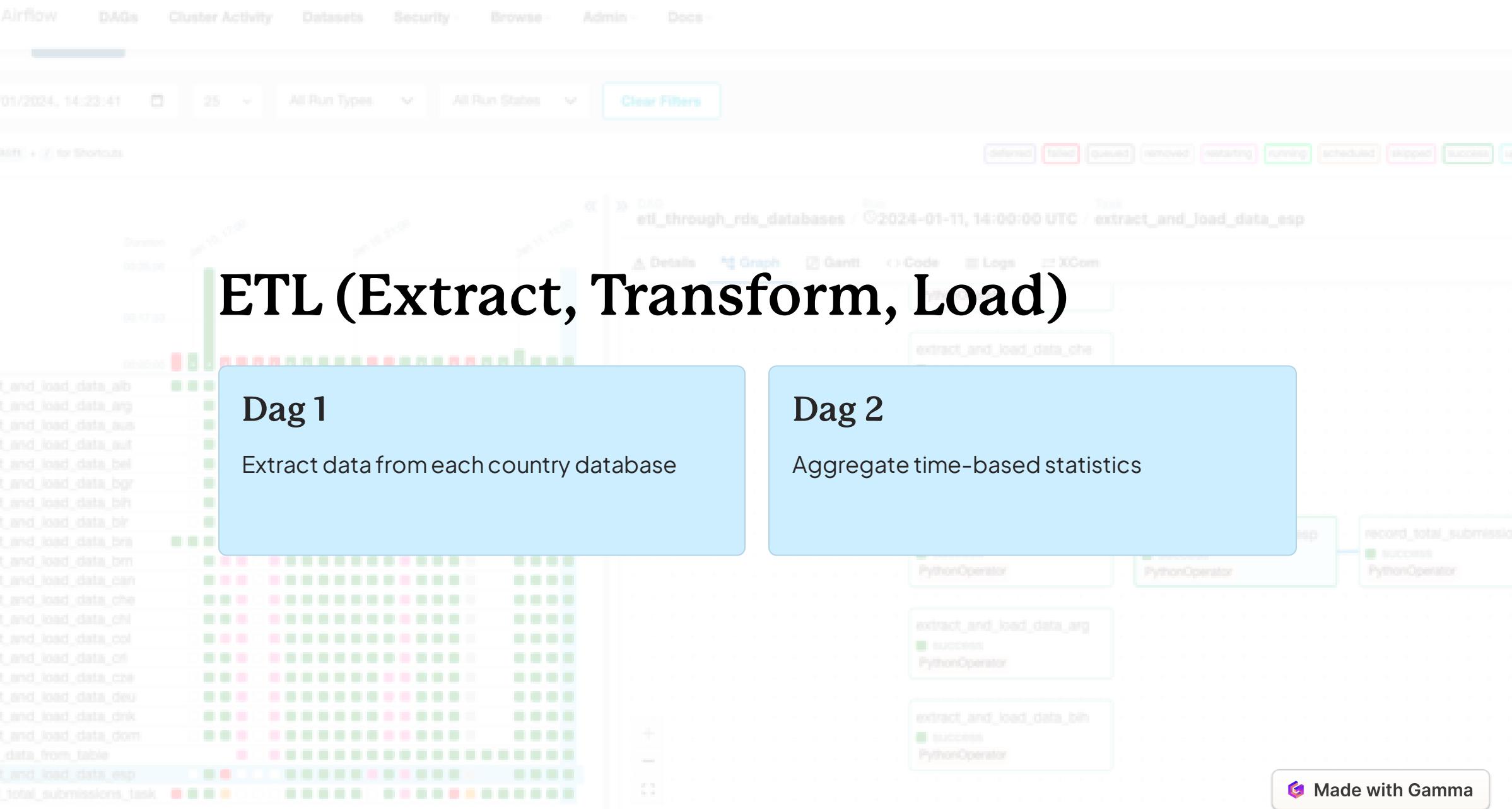
Data structure

This chart expects the following sort of data structure:

```
{  
  "datasets": [  
    // one of the below object for each country  
    {  
      "id": "A three-letter country code, uppercased. e.g. GBR",  
      "data": [  
        {  
          "x": "DURECEC as an integer, e.g. 6",  
          "y": "BELONG as a float, e.g. 1.1",  
          "submissions": "a count of submissions from this country as an integer, e.g. 412"  
        }  
      ]  
    }  
  ]  
}
```

Solution Strategy





Flask API

```
@app.route('/early_education_and_belonging')
def early_education_and_belonging():
    conn = get_db_connection()
    cur = conn.cursor()
    datasets = {'datasets' : None}
    cur.execute('''
        WITH int_test AS(
            SELECT cnt,cast(durecet AS integer), cast(belong AS float)
            FROM test
            WHERE durecet != 'NA' AND belong != 'NA')
        SELECT cnt AS id, ROUND(AVG(durecet),0) AS x, AVG(belong) AS y, COUNT(cnt) AS submissions
        FROM int_test
        GROUP BY cnt
        ORDER BY cnt
    ''')
    results = cur.fetchall()
    formatted_data = []
    for result in results:
        country_data = {
            "id": result["id"],
            "data": [
                {
                    "x": int(float(result["x"])), # Convert x to an integer
                    "y": float(result["y"]), # Keep y as a float
                    "submissions": int(result["submissions"]) # Convert submissions to an integer
                }
            ]
        }
        formatted_data.append(country_data)
    final_output = {"datasets": formatted_data}
    cur.close()
    conn.close()
    return final_output

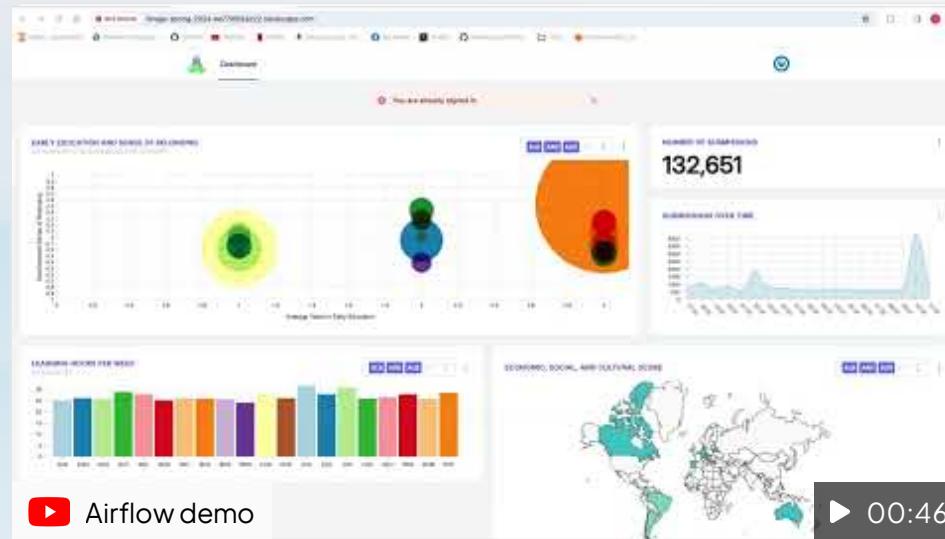
if __name__ == '__main__':
    context = ("cert.pem", "key.pem")
    app.run(debug=True, port = 5000, host='0.0.0.0', ssl_context=context)
```

```
        ")
results = cur.fetchall()
formatted_data = []
for result in results:
    country_data = {
        "id": result["id"],
        "data": [
            {
                "x": int(float(result["x"])). # Convert x to an integer
```

Flask API

- 5 endpoints
- URL access
- Https get request
- Security / SSL
- Json output

Results





Areas of Improvement and next steps

- 1 Use stream processing
- 2 Explore other solutions