

Situation

Client

Harmonyland is a blessed country, led by an affable and clear-sighted ruler. He takes great pride in its effort to bring peace, happiness, and harmony to all its citizens.

To do so, the benevolent ruler heavily rely on his fellow harmonymakers. A governmental agency dedicated to spread harmony around the country. To reach their ambition, they bring assistance to any agitated person and help them to recover peace and harmony. More generally they help citizen to stay in line with their country harmonious goal

To help its harmonymakers squads, Harmonyland engineers have created a working autonomous drone called harmonywatcher.

They need you to create the program that will receive and manage harmonywatchers's data.

This program must

- store every harmonywatcher data
- trigger alerts
- enable harmonymaker officers to perform analysis on harmonywatcher data

Drone description

Each harmonywatcher sends a report every minute.

One report contains

- harmonywatcher id
- harmonywatcher current location (latitude, longitude)
- name of surrounding citizens (identify with facial recognition) with their computed «harmonyscore». Harmonyscore being recomputed for every report.
- words heard by the watcher in its surrounding

Alert

When a citizen harmonyscore is bad, your program must trigger an alert with the location of the harmonywatcher and the name of the agitated citizen.

Peacemakers will take it from there and help the person to find peace and harmony.

They may send him to a harmonycamp. In such camp citizen learn to reach happiness following the ideas of the benevolent leader. Or they will put him in a sustainable and never ending peace state.

This alert must be triggered as quickly as possible because an agitated citizen may spread its lack of harmony to other citizens. Thus, the harmonymaker reaction must be as fast as possible.

Statistics

Harmonymakers are convinced that we need to keep every harmonywatcher report in order to make statistics and improve their Harmonystate harmony. But they still don't know what kind of question/statistic they will want to address. Harmonystate engineer estimate that when the first wave of harmonywatcher will be operational the sum of all their daily report will weight 200Gb. They also estimate that less than 1% of harmonywatcher report contains alert.

Failed attempt

To create a POC of the program, Harmonystate hired a team of data-scientists and. Despite all their efforts, this team have not been able to set up a scalable program that can handle the load.

Preliminary questions

- 1) What technical/business constraints should the data storage component of the program architecture meet to fulfill the requirement described by the customer in paragraph «*Statistics*» ?
So what kind of component(s) (listed in the lecture) will the architecture need?
- 2) What business constraint should the architecture meet to fulfill the requirement describe in the paragraph «*Alert*»? Which component to choose?
- 3) What mistake(s) from Harmonystate can explain the failed attempt?
- 4) Harmonystate has likely forgotten some technical information in the report sent by the drone. In the future, this information could help Harmonystate make its harmonywatchers much more efficient. Which information?

Project

Harmonystate understands this is beyond their team limits, it can not put in place a programm to deal with the drone's data. Harmonystate asks you for advice to design an architecture allowing them to create a product they could sell to different police forces.

It's up to you to report and recommend the right architecture.

Based on the preliminary questions, your solution is very likely to include :

- at least one distributed storage
- at least one distributed stream
- at least two stream consumer