

UBER

Uber's mission is to provide transportation that is as reliable as running water. Achieving that mission will require world class engineers and a lot of data.

Background

Our team currently manages a large distributed Cassandra NoSQL database service that is used by many teams throughout Uber. The primary method of querying this data is based on a "pull" model--any client wishing to see the latest version of the data must request the data from the database.

While viewing the data, it is often necessary for the client to know when data has changed. This can be accomplished by pulling data frequently (thus reducing the possibility that anything has changed), but it places a much higher load on the database system. Pulling data less frequently reduces the load on the database system, but significantly increases the chance that the client is looking at data that is no longer valid.

Project Description

To improve the performance of the service, we want to create a "push and subscribe" model on top of the database using a distributed Kafka queue.

Under this model, when the data changes, the database service would push a small amount of metadata to a Kafka queue. At the same time, clients will subscribe to this queue, which will allow them to be instantly alerted to changes in the database when they happen. This will allow clients to re-query the database only when it is guaranteed that they will get updated data back.

A finished version of the project will include a fully functioning proof-of-concept, with code that can be quickly dropped into the existing frameworks at Uber.

Requirements

We are looking for a team of 3 (possibly up to 4) engineering students. Students should be confident programming in Java and should feel very comfortable working in Linux.

Most work will need to be done at the Uber office in Louisville, CO, and all team members will be required to sign an NDA.

Offers for a summer internship are possible if students show strong talent and a fierce determination to solve problems.