Description

Your task is to implement an aggregation/ingestion system for logging events. There will be a client that sends randomly generated logging events to a server that accepts the events and pushes them on to Kafka. Another server consumes the log events and writes them to a Cassandra database.

You will have to define what a log event is, how to generate them randomly, what is the API for the client-server interaction, and the Cassandra schema.

Everything will be run from the command line; we assume a Unix-like environment (e.g. Linux, macOS).

Technologies

1. Scala or Java as the programming language. We use Scala internally but if you are not comfortable with it, please use Java. It does not matter for the evaluation of this task which language you will choose.
2. Apache Thrift for the definition of the logging event schema and the needed client-server interaction API.
3. Kafka as the intermediate destination of the logging events.
4. Cassandra as the final destination of the logging events.
5. Maven or SBT for the project build manager.
6. Bash for shell scripts.

Logging event

The minimum set of attributes are:

- `v` of type `i16` (this is a thrift type), which indicates a version number for the logging event schema. E.g. we expect this to have the value of `1` for this programming task.
- `time` which represents the time of the logging event; please select an appropriate type and explain the decision.
- `m`, which is the logging message itself; please select an appropriate type.

Could you think of a couple more fields that could be of help in a more realistic system? Feel free to implement extra fields, explaining their purpose.

Cassandra Schema

No specific requirement but please explain your decisions.

Products

The "source" archive is the primary product of this task and includes the source code, any other supporting files that may be needed, and a README that explains how to produce the "binary".

The "binary" archive is the secondary product. You do not need to provide it but it should be clear how to produce it from the "source". In particular, it is a distribution archive that contains these modules, each one in its own folder:

1. Module `thrift-client` is the program that we use to send the logging event to `thrift-server`.
2. Module `thrift-server` implements the other end of the thrift API and sends logging events to Kafka.
3. Module `kafka-consumer` reads events from Kafka and writes them to Cassandra.

Each module is self-contained and includes everything needed in order to run the respective code. Please use Bash shell scripts to run the JVM code.

For anything not specified here (e.g. Kafka installation, Cassandra installation), please state how we can reproduce the corresponding running environment.

Please send us the distribution and source code archive URLs.

Software versions

- Scala (if chosen): 2.11.x
- JDK: 1.8
- Thrift: please choose
- Kafka: please choose
- Cassandra: please choose
- Shell: Bash 4

Important

Feel free to ask clarification questions, until you are confident you can start working on this task. Then, please provide us with an estimation of how much time you will need for the complete implementation. We need to know a) how much programming time you estimate that it will take you and b) an end date when we should expect the archive URLs mentioned in the Products section.

This is a volunteering effort and we respect your time. The estimation is not part of the evaluation, it is just for statistics.