Systems for Data Science - HW7

S4 - Paper Review

Shubham Shetty

March 02 2021

1 Summary

S4 (Simple Scalable Streaming System) is a general purpose, real-time, distributed, decentralized, partially fault tolerant, scalable, event-driven and pluggable platform which allows programmers to handle and process real-time streaming data. It was developed by Yahoo! for the purpose of search advertising and handling user sessions in real-time. Streaming data model is a model where the data is available continuously and should be processed immediately to ensure best utilisation. In contrast to static data model, in streaming data model the data is brought to the queries whereas in the former it is the queries which are brought to the data.

S4 is able to process information from multiple event streams parallely using data driven models in real time with low latency. The various application areas for S4 include network monitoring, web personalization and user session modelling, online real-time machine learning, etc. S4 allows programmers to easily write applications for processing data streams.

The S4 platform is built on a distributed architecture and can work on several commodity hardware units. The core of the S4 platform is the actors programming model. In this model, each processing unit can communicate with every other unit in the system. This peer-to-peer model ensures that the system is decentralized and there is no single point of failure. The platform is also pluggable and science-friendly.

The main processing within the S4 platform is done by Processing Elements (PEs). S4 consumes an incoming event data stream input, and then passes this stream to a PE. A load balancer ensures that the processing is divided equally across all PEs in the system. After processing the stream PEs can either emit one or more event streams which can be consumed by other PEs, or can publish the output results. ZooKeeper coordinates communication between the nodes.

2 Strengths

The strengths of the S4 system are -

- S4's Streaming programming model is a graph-based programming model. This means it is easier to visualise and more intuitive.
- S4 code is highly reusable. As most streaming functionalities are common, a lot of code can be reused for various applications.
- Parallelism is handled by the platform. Users/Programmers do not have to worry about implementing multi-threading via code.
- As S4 computes data on the fly (data is stored in-memory) and does not store the streams, it avoids any storage or disk access bottlenecks.

3 Unclear Aspects

The unclear aspects regarding the S4 system are -

- S4 system uses load balancing - how effective is the load balancer as the system scales out?
- Is the Actor's programming model based on peer-to-peer systems? How does it work?
- How does S4 compare to other stream processing platforms such as Apache Kafka, AWS Kinesis, Spark Streaming etc.

4 Limitations/Areas of Improvement

The areas of improvement, or limitations of the S4 system are -

- Can include dynamic load balancing to improve elastic scalability.
- S4 platform should introduce drivers to communciate with systems using non-Java based languages like C/C++ or Python.
- S4 should have inherent query functionality.
- Can have an improved communication layer, which is more optimised and reliable for control messages.