Performance of Splunk for the TDAQ Information Service at the ATLAS experiment

Yoshiji Yasu and Andrei Kazarov on behalf of the TDAQ collaboration

I. Introduction

The ATLAS Trigger and Data Acquisition (TDAQ) is a large, distributed system composed of several thousand interconnected computers and tens of thousands software processes. It is important for detecting the errors and understanding the system performance to monitor the status of the computers and the software processes at the TDAQ operation. The studies of the software to ease the system analysis had been done. Any system implementing these functions have to be flexible in order to adapt to the amount of data produced and requested by the users for analysis and visualization.

Splunk is a candidate that has the desired functionality. Splunk is a commercial product produced by Splunk Inc. and a general-purpose search, analysis & reporting engine and a distributed, composable, semi-structured database for time-series text data (typically machine data in large-scale data processing).

The functionality of Splunk has been evaluated and the scalability of indexing and querying has been studied. The querying includes searching, aggregation, correlation and visualization.

II. Information Service and Splunk

The multiple sources to be monitored are in the Information Service (IS). All TDAQ applications publish their monitoring data into the IS servers.

Splunk reads information from IS servers in real-time and at runtime and use multiple Forwarders and Indexer component to put it to the persistent storage (green arrows). Indexers are also used for querying information from Splunk (orange arrow) where request is sent from a Web Browser via the Search Head component.

Here is an example of the events. Any message starts with "Splunk..." followed by the index name. The event consists of two lines. The second line contains the timestamp and the value of the attributes.

```
# Splunk...| index=...| time=...| attribute=value
```

III. Query Dashboards

The dashboards developed for the TDAQ operation are shown here. The dashboard provides an at-a-glance view for the search result.

The following is a general-purpose dashboard for the aggregation and the correlation of the indexed data. The user selects index name first. After choosing the index, the source and the host can be selected. There are three plots. First and second plots show the average of attribute specified as a function of time series. Last one is for the correlation.

IV. Performance Measurements

A. Indexing Performance

The indexing rates were measured. There are several different tests where number of indexers varies from 1 to 8.

- There are 6 Splunk forwarders. The multiple forwarders increases the forwarding rate linearly.
- The indexing is CPU-bound in our case. As far as the CPU load of the indexer is not saturated at indexing time, the indexing rate is expected to scale linearly.

B. Querying Performance

The querying time depends on the query type and the number of events that are actually accessed by the indexers.

- The querying time scaled linearly with a function of the number of events searched.

The execution times of 4 queries that run simultaneously were measured to investigate whether multiple queries decrease their performance or not. The queries are listed as follows:

```
Query 1: index=...| time=...| attribute=value
Query 2: index=...| time=...| attribute=value
Query 3: index=...| time=...| attribute=value
Query 4: index=...| time=...| attribute=value
```

The indexing is CPU-bound in our case. As far as the CPU load of the indexer is not saturated at indexing time, the indexing rate is expected to scale linearly.

V. Conclusions

- The indexing scaled linearly with the number of indexers.
- An indexer had typically ~200% CPU usage.
- The querying scaled linearly with the number of events searched.
- A query running for a long time had typically ~100% CPU usage.
- If the CPU resource is limited, multiple queries can run simultaneously during the indexing without decreasing the performance.
- When we design a system using Splunk, we need to estimate not only the number of indexer, forwarder and search head, but also the value of the parameter "volume used today" and the actual size of Splunk database increased in a day. If the CPU resource is limited, the number of search run concurrently can be restricted by Splunk configuration.
- If the forwarding rate is too large in average, the data will be overflowed at forwarders, but the indexing is CPU-bound in our case. As far as the CPU load of the indexer is not saturated at indexing time, the indexing rate is expected to scale linearly.

Acknowledgment

The authors would like to thank Giovanna Lehmann Mictic and Igor Soloviev for leading our project. They would also like to thank Alexsandro D. Sioce for his effort to evaluate Splunk which our job is based on, as well as Werner Vandelie, Luca Magnoni and Alejandro Iribarren for their support.