Developing and Optimizing applications in Hadoop

Prasanth Kothuri, Daniel Lanza Garcia, Joeri Hermans
CERN IT Database Group
Outline

- What is Hadoop?
- Data Ingestion
- Data Model & Data Formats
- Hadoop Processing Frameworks
  - Spark
- Batch / request-response application
- Troubleshooting
Hadoop

• A framework for large scale data processing
  • Distributed storage and distributed processing
  • Shared nothing architecture – scales horizontally
  • Optimized for high throughput on sequential data access
Data Ingestion - What are the challenges?

- Variety of data sources
  - Databases
  - Web
  - REST
  - Logs

- HDFS is a file system, not a database
  - You need to store files

- There is always a **latency** when storing to HDFS
  - data streams has to be materialized in files
  - creating a file per a stream event will kill HDFS, Hadoop works efficiently for big files
  - events has to be written in groups
Data Ingestion - Flume

- High-volume ingestion into Hadoop of event-based data
- Large library of *sources* and *sinks* cover all the bases of *what* to consume and *where* to write
- Highly flexible, configurable and containable memory footprint
- Example
Data Ingestion – Flume Extended

• Developed two additional new sources
  • JDBCSource
    • Able to consume data from database tables
  • LogFileSource
    • Able to consume data from log files

• Graceful restart of the agent
  • By preserving the last processed event based on timestamp or any other column

• Ability to identify the duplicate events
  • By holding a list of hashes of events
Data Ingestion - Flume deployment

✓ Stable & Works well
✓ Contextual routing
✓ Overcomes small files problem

× high availability, replication and retention
× Sinking to multiple data stores
Data Ingestion - Kafka

- Distributed messaging system
  - High availability of events
  - Events are partitioned and replicated across multiple nodes
  - Scalable, fault tolerant and durable
  - Pull-based system
    - Events are retained for a set amount of time
    - Consumers dictate the pace
    - Aggressive batching of events

- Benchmarking done on CERN OpenStack Infrastructure
  - 3 node Kafka Cluster with Zookeeper installed on separate VM
  - VM Spec (m2.large) : 4 VCPUs, 7.3 GB RAM & 100G storage
Flume is very flexible, however high availability, scale and guarantees can only be achieved with Apache Kafka.
Data Formats

• Data Formats are an important aspect for optimizing storage footprint and scan (query) performance
  ➢ Text Formats
    • TEXT, CSV, JSON
    • You waste lot of CPU cycles parsing JSON
  ➢ Binary container formats
    • Avro
      • Compact, fast, binary format
    • Parquet
      • Column oriented data serialization format optimized for high compression and high scan efficiency
Criteria for choosing a Data Format

1. **Data Ingestion speed** – the time it takes to write data onto storage
2. **Sequential access** – scanning through the entire dataset
3. **Analytics** – Aggregations using group by (column projection and predicate push down)
4. **On-Disk Storage footprint** – the amount of space the dataset occupies on the storage
5. **Random access** – look ups, although this depends more on the partitioning of your datasets
Comparison of AVRO vs PARQUET

We choose parquet for scan performance and analytical queries.
Apache Spark

- Open source, large and active use base

- Wide library support for
  - unstructured input data
  - efficient analysis storage formats
  - stats and machine learning algorithms

- provides parallel processing primitives
  - declarative - traditional SQL queries
  - imperative (no-SQL)

- bindings to most popular analysis languages: Python, R, Scala, Java
Apache Spark

- Spark streaming for real time alerting
- Spark core (batch processing) for user facing analytics
- Low latency access with Spark Thrift server
User Interface: SPARK -> ElasticSearch/Kibana

- Application users access reports using Kibana – an open source visualization tool
- Reports (aggregations) are computed using Spark and delivered to Kibana

```scala
/* Write to ES from Spark(scala) */
import org.elasticsearch.spark.sql._
val sen_p=sqlContext.read.parquet("/path/to/HDFS/file")
sen_p.registerTempTable("sensor_ptable")
sqlContext.sql("SELECT ts, element_id, count(*) as cnt FROM stable group by ts,element_id") \
  .saveToEs("sensor/metrics")
```
Hadoop performance troubleshooting

- **hprofile**
  - Tool developed to troubleshoot application performance on Hadoop
  - Ability to identify part of the code the application is spending most time on and visualize this in a human readable manner using flamegraphs

- **Usage and more information**
  - `sh hprofiler.sh -f 300 -t 60 -c [cluster address] -j "grep 123456789" -o results`
  - [https://github.com/cerndb/Hadoop-Profiler](https://github.com/cerndb/Hadoop-Profiler)
Hadoop performance troubleshooting

• This profiler helps to identify the performance bottlenecks in distributed applications
Final Application Architecture

Flume Agents -> kafka

Real-time

Batch processing

Spak Streaming

HADOOP

Web UI

Lambda Architecture
Conclusion

• Data Ingestion, formats and processing framework are key aspects of building Hadoop Application

• Out of the myriad of Hadoop tools available, it is possible to build Hadoop Application using Kafka, Parquet and Spark

• ElasticSearch / Kibana can be leveraged to deliver dashboards

• Challenge of troubleshooting distributed application can be overcome to some extent using tools like hprofiler
Hadoop Service at CERN

- **Service** provided by CERN-IT for Experiments and CERN users
- Projects ongoing with Experiments, Accelerators sector and IT
- Hadoop Users Forum for open discussions: subscribe to egroup `it-analytics-wg`
- Getting started material: Hadoop tutorials [https://indico.cern.ch/event/546000/](https://indico.cern.ch/event/546000/)

- Related talks/posters at CHEP 2016:
  - A study of data representations in Hadoop to optimize data storage and search performance of the ATLAS EventIndex, poster on Tuesday 16:30
  - Hadoop and friends - first experience at CERN with a new platform for high throughput analysis steps, talk on Thursday at 14:45
  - Integration of Oracle and Hadoop: hybrid databases affordable at scale, talk on Monday at 10:45