Contents

Problem statement

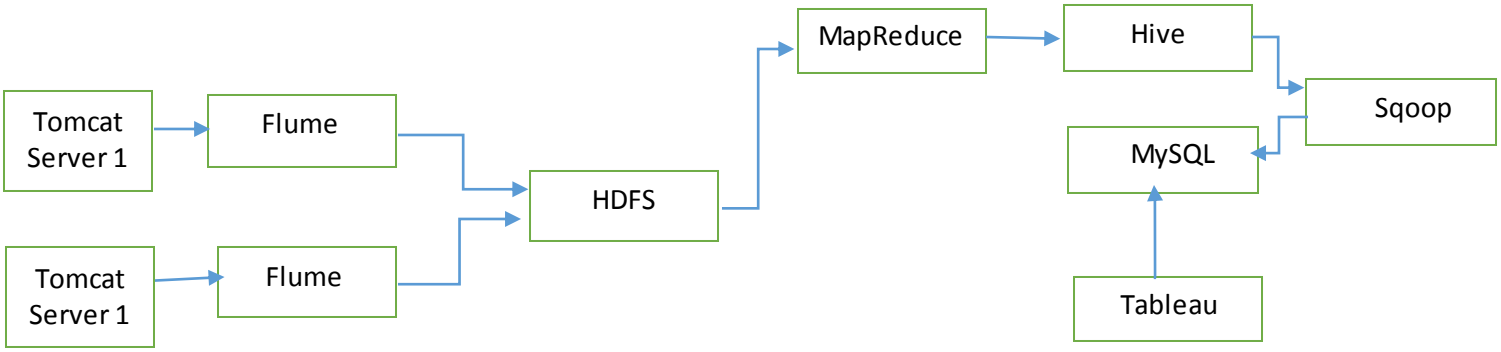
The growth of websites and the Internet has opened up new research, social, entertainment, education and business opportunities. With the fast growth of the Internet, a massive amount of data generated by Web Server.

We this massive amount of data, we are going to analyse some of trends. Here we are analysing E-Commerce website Web Server Logs.

Steps for analysing log files are follows:

1. Load data into HDFS using HDFS client
2. Develop MapReduce program to parse logs and convert request string into structured format (/a/b/c/d => a b c d)
21.125.155.111 - - [01/Jan/2012:12:07:48 +0530] "GET /digital-cameras/digital-camera/sony-qx-dsc-qx100-point-shoot-digital-camera-black.html HTTP/1.1" 200 1470 "Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.2.17) Gecko/20110420 Firefox/3.6.17" "-"
3. Count of page views by individual user
4. Top / Bottom 5: catagery-1/ catagery-2 / page /users / entry pages (Exclude status code other than 200, also exclude record related to css/js/image)
5. Total page views / Category wise page views / unique page views
6. Count of status code = 200 / 404 / 400 / 500
7. Load results into tables in MySQL database using Sqoop.

Solution Architecture



Software/Tools Specification

As shown in the above architecture there are various components used in Log Analysis.

Flume: Collection streaming log data into HDFS from various Tomcat Servers.

HDFS: HDFS is the storage file system for huge volumes of log data collected by flume.

MapReduce: MapReduce parses HDFS files and convert into relevant fields needed for analysis.

Hive: Hive will define schema to this structured data and schema will be stored in hive metastore. We write query on top of hive and resulting data stored in HDFS.

Sqoop: Sqoop used to import HDFS data into MySQL database.

MySQL: The resulted data stored in MySQL

Tableau: It is a visualization tool that provides connectivity to MySQL store. So we may present user stories to clients.

Solution Description: Steps for processing and analysing log data

Input

Some samples from Apache Tomcat logs.

```
21.125.155.111 - - [01/Jan/2012:12:07:48 +0530] "GET /digital-cameras/digital-
camera/sony-qx-dsc-qx100-point-shoot-digital-camera-
black.html HTTP/1.1" 200 1470 "Mozilla/5.0 (Windows; U; Windows NT 6.1; en-
US; rv:1.9.2.17) Gecko/20110420 Firefox/3.6.17" "-"
168.42.128.252 - - [01/Jan/2012:12:09:36 +0530] "GET /digital-cameras/digital-
camera/canon-powershot-sx50-hs-point-shoot-
camera.html HTTP/1.1" 200 195 "Mozilla/5.0 (Windows; U; Windows NT 6.1; en-
US; rv:1.9.2.17) Gecko/20110420 Firefox/3.6.17" "-"
196.34.35.201 - - [01/Jan/2012:12:18:18 +0530] "GET /tvs-audio/blu-ray-dvd-
players/d-m-holdings-inc-denon-dbt-1713ud-blu-ray-
player.html HTTP/1.1" 200 1503 "Mozilla/5.0 (Windows NT 6.2) AppleWebKit/537.1
7 (KHTML, like Gecko) Chrome/24.0.1312.56 Safari/537.17" "-"
91.228.209.0 - - [01/Jan/2012:12:28:04 +0530] "GET /home-
appliances/fans/reconnet-rhcfg-1201-ceiling-
fan.html HTTP/1.1" 200 773 "Mozilla/5.0 (Windows NT 6.2) AppleWebKit/537.17 (K
HTML, like Gecko) Chrome/24.0.1312.56 Safari/537.17" "-"
```

Flume Commands

Flume used as DataStream Collector for collecting events from log server. Taildir watch the specified files, and tail them in nearly real-time once detected new lines appended to the each files. If the new lines are being written, this source will retry reading them in wait for the completion of the write.

This source is reliable and will not miss data even when the tailing files rotate. It periodically writes the last read position of each files on the given position file in JSON format. If Flume is stopped or down for some reason, it can restart tailing from the position written on the existing position file.

`flume-weblogs.conf`

```
agent.sources = tomcat
agent.sinks = hdfs-sink
agent.channels = memory-channel
agent.sources.tomcat.type = TAILDIR
agent.sources.tomcat.filegroups = f1
agent.sources.tomcat.filegroups.f1 = /home/devendrashrivardhankar1451/project/
weblogs/weblogs_10_lakh_rec.txt
agent.sinks.hdfs-sink.type = hdfs
agent.sinks.hdfs-
sink.hdfs.path = /user/devendrashrivardhankar1451/project/flume/10_lac_rec/log
s
agent.sinks.hdfs-sink.hdfs.fileType = DataStream
agent.sinks.hdfs-sink.hdfs.writeFormat = Text
agent.sinks.hdfs-sink.hdfs.batchSize = 200
agent.sinks.hdfs-sink.hdfs.rollSize = 73400320
agent.sinks.hdfs-sink.hdfs.rollCount = 0
agent.channels.memory-channel.type = memory
agent.channels.memory-channel.capacity = 1000
agent.channels.memory-channel.transactionCapacity = 500
agent.sources.tomcat.channels = memory-channel
agent.sinks.hdfs-sink.channel = memory-channel
```

Flume uses default `taildir_position.json` to keep track of last read lines from source. We have to explicitly create this files else flume will throw error.

```
cd ~/.flume/
```

```
cat taildir_position.json
```

```
flume-ng agent -n agent -c conf -f
/home/devendrashrivardhankar1451/project/flume-weblogs.conf
```

MapReduce Commands

We have created MR program to parse log files in structured format.

We are only interested in following fields.

IP Cat1 Cat2 Status_Code URL

LogParser.java

```
package com.dev.mr;

import java.io.IOException;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

public class LogParserDriver {

    public static void main(String[] args) throws IOException, ClassNotFoundEx
ception, InterruptedException {

        if (args.length < 1) {
            System.err.println("Please provide HDFS directory");
            System.exit(1);
        }
        System.out.println("MapReducer Job Started");
        Configuration configuration = new Configuration();
        Job job = Job.getInstance(configuration, "log-parser");
        job.setInputFormatClass(TextInputFormat.class);
        job.setOutputFormatClass(TextOutputFormat.class);

        job.setOutputKeyClass(NullWritable.class);
        job.setOutputValueClass(Text.class);

        job.setJarByClass(LogParserDriver.class);
        job.setMapperClass(LogMappper.class);
        job.setNumReduceTasks(0);

        FileInputFormat.addInputPath(job, new Path(args[0]));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));

        System.out.println(job.waitForCompletion(true));
        System.out.println("MapReducer Job Completed");
```

```
    }
}
```

LogMapper.java

```
package com.dev.mr;

import java.io.IOException;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

public class LogMappper extends Mapper<Object, Text, NullWritable, Text> {
    @Override
    protected void map(Object key, Text value, Mapper<Object, Text, NullWritab
le, Text>.Context context)
            throws IOException, InterruptedException {
        String ip = null;
        String category1 = null;
        String category2 = null;
        String status_code = null;
        String url = null;
        String ouputStr = null;
        String record = value.toString();

        String logRegex = "^([\\d.]+) (\\S+) (\\S+) \\[([\\w:/]+\\s[+\\-
]\\d{4})\\] \"(.+?)\" (\\d{3}) (\\d+) \"([^\"]+)\" \"([^\"]+)\"";
        String urlRegex = "(\\S+) (\\S+) (\\S+)";

        Pattern pattern = Pattern.compile(logRegex);
        Matcher matcher = pattern.matcher(record);
        Pattern urlPattern = Pattern.compile(urlRegex);
        if (matcher.find()) {

            ip = matcher.group(1);
            status_code = matcher.group(6);
            String pageULR = matcher.group(5);
            Matcher urlMatcher = urlPattern.matcher(pageULR);
            if (urlMatcher.find()) {
                String catURLS = urlMatcher.group(2);
                String[] split = catURLS.split("/");
                for (int i = 0; i < split.length; i++) {
                    if (i == 1)
```

```
                        category1 = split[1];
                    if (i == 2)
                        category2 = split[2];
                    if (i == 3)
                        url = split[3];
                }

            }
            ouputStr = ip + "," + category1 + "," + category2 + "," + status_c
ode + "," + url;
        }
        context.write(NullWritable.get(), new Text(ouputStr));
    }

}
```

```
hadoop jar /home/devendrashrivardhankar1451/project/LogAnalysis.jar com.dev.mr
.LogParserDriver project/flume/10_lac_rec/logs project/mapreduce/10_lac_rec/lo
gs
```

Hive Commands

Once we parse Logs in Structured format, we create Hive Metadata and import data from HDFS to Hive.

```
CREATE TABLE IF NOT EXISTS weblog_analysis(
ip STRING COMMENT 'IP Address of the User',
cat1 STRING COMMENT 'Product Category',
cat2 STRING COMMENT 'Product SubCategory',
status_code INT COMMENT 'Response Code from Website',
page_url STRING COMMENT 'HTML Link')
COMMENT 'This is the apache access log table'
PARTITIONED BY(dt STRING)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
STORED AS TEXTFILE;
```

```
LOAD DATA INPATH '/user/devendrashrivardhankar1451/project/mapreduce/10_lac_re
c/logs' OVERWRITE INTO TABLE weblog_analysis PARTITION(dt='2020-05-07');
```

We created Hive User Defined Function to filter out css/js/image record from URL.

URLValidator.java

```
package com.dev.hive;

import org.apache.hadoop.hive.ql.exec.UDF;
import org.apache.hadoop.io.Text;

public class URLValidator extends UDF {
    String[] items = { "css", "js", "image" };

    public Text evaluate(Text input) {
        if (input == null) {
            return null;
        }
        for (String item : items) {
            if (input.toString().contains(item)) {
                return null;
            }
        }
        return new Text("found");
    }
}
```

```
add jar /home/devendrashrivardhankar1451/project/URLValidator.jar;
create temporary function urlvalidator as 'com.dev.hive.URLValidator';
```

Analysis

Now onwards we are going to answer some of questions.

Solution template involve following sequential commands:

1 HiveQL to analyse data

2 MySQL table creation

3 Sqoop export command to import data from HDFS to MySQL

4 MySQL result

Count of page views by individual user

```
INSERT OVERWRITE DIRECTORY '/user/devendrashrivardhankar1451/project/hive/resu
lts/3'
row format delimited fields terminated by ','
select ip, count(*) from weblog_analysis group by ip;

create table dev_result3
(
    ip varchar(15),
    count int
);

sqoop export --connect jdbc:mysql://cxln2.c.thelab-240901.internal/sqoopex  --
table dev_result3  --export-
dir /user/devendrashrivardhankar1451/project/hive/results/3 --
username sqoopuser --password ****** --fields-terminated-by ','
```

```
mysql> use sqoopex
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from dev_result3 limit 20;
+--------------+-------+
| ip           | count |
+--------------+-------+
| 0.0.134.43   |     1 |
| 0.0.17.106   |     1 |
| 0.0.229.150  |     1 |
| 0.0.250.189  |     1 |
| 0.0.65.145   |     1 |
| 0.0.83.37    |     1 |
| 0.1.11.208   |     1 |
| 0.1.113.252  |     1 |
| 0.1.119.30   |     1 |
| 0.1.124.157  |     1 |
| 0.1.129.46   |     1 |
| 0.1.160.14   |     1 |
| 0.1.178.176  |     1 |
| 0.1.21.120   |     1 |
| 0.1.67.159   |     1 |
| 0.1.73.60    |     1 |
| 0.1.82.156   |     1 |
| 0.10.105.6   |     1 |
| 0.10.139.70  |     1 |
| 0.10.153.201 |     1 |
+--------------+-------+
20 rows in set (0.00 sec)
```

Top 5: catagery-1/ catagery-2 / page /user (Exclude status code other than 200, also exclude record related to css/js/image)

```
INSERT OVERWRITE DIRECTORY '/user/devendrashrivardhankar1451/project/hive/resu
lts/4/1'
row format delimited fields terminated by ','
select cat1, count(*) total from weblog_analysis where status_code=200 and url
validator(page_url) is not null group by cat1 order by total desc limit 5;

create table dev_result41
(
    cat1 varchar(100),
    total int
)

sqoop export --connect jdbc:mysql://cxln2.c.thelab-240901.internal/sqoopex  --
table dev_result41  --export-
```

```
dir /user/devendrashrivardhankar1451/project/hive/results/4/1  --
username sqoopuser --password ****** --fields-terminated-by ','
```

```
mysql> select * from dev_result41;
+-----------------+--------+
| cat1            | total  |
+-----------------+--------+
| tvs-audio       | 196543 |
| digital-cameras | 105694 |
| home-appliances | 317560 |
| computers       | 211404 |
| mobiles         | 147713 |
+-----------------+--------+
5 rows in set (0.00 sec)
```

```
INSERT OVERWRITE DIRECTORY '/user/devendrashrivardhankar1451/project/hive/resu
lts/4/2'
row format delimited fields terminated by ','
select cat2, count(*) total from weblog_analysis where status_code=200 and url
validator(page_url) is not null group by cat2 order by total desc limit 5;

create table dev_result42
(
    cat2 varchar(100),
    total int
);

sqoop export --connect jdbc:mysql://cxln2.c.thelab-240901.internal/sqoopex  --
table dev_result42  --export-
dir /user/devendrashrivardhankar1451/project/hive/results/4/2  --
username sqoopuser --password ****** --fields-terminated-by ','
```

```
mysql> select * from dev_result42;
+------------------------+--------+
| cat2                   | total  |
+------------------------+--------+
| laptops                |  70765 |
| smart-phones           | 119688 |
| air-conditioner-coolers |  91687 |
| tablets                |  91332 |
| washing-machine        |  70568 |
+------------------------+--------+
5 rows in set (0.00 sec)
```

```
INSERT OVERWRITE DIRECTORY '/user/devendrashrivardhankar1451/project/hive/resu
lts/4/3'
row format delimited fields terminated by ','
select page_url, count(*) total from weblog_analysis where status_code=200 and
 urlvalidator(page_url) is not null group by page_url order by total desc limi
t 5;

create table dev_result43
(
    Page_url varchar(300),
    total int
);

sqoop export --connect jdbc:mysql://cxln2.c.thelab-240901.internal/sqoopex  --
table dev_result43  --export-
dir /user/devendrashrivardhankar1451/project/hive/results/4/3  --
username sqoopuser --password ****** --fields-terminated-by ','
```

```
mysql> select * from dev_result43;
+-----------------------------------------------+-------+
| Page_url                                      | total |
+-----------------------------------------------+-------+
| NULL                                          | 35571 |
| reconnect-rhpfg6001-pedestal-fan.html         |  7279 |
| amazon-kindle-e-reader-wi-fi-6-inch.html      |  7178 |
| jbl-pulse-speaker-black.html                  |  7206 |
| swipe-fablet-6-android-smart-phone-white.html |  7191 |
+-----------------------------------------------+-------+
5 rows in set (0.01 sec)
```

```
INSERT OVERWRITE DIRECTORY '/user/devendrashrivardhankar1451/project/hive/resu
lts/4/4'
row format delimited fields terminated by ','
select ip, count(*) total from weblog_analysis where status_code=200 and urlva
lidator(page_url) is not null group by ip order by total desc limit 5;

create table dev_result44
(
    ip varchar(15),
    total int
);

sqoop export --connect jdbc:mysql://cxln2.c.thelab-240901.internal/sqoopex  --
table dev_result44  --export-
dir /user/devendrashrivardhankar1451/project/hive/results/4/4  --
username sqoopuser --password ****** --fields-terminated-by ','
```

```
mysql> select * from dev_result44;
+-----------------+-------+
| ip              | total |
+-----------------+-------+
| 107.85.179.52   |     2 |
| 1.94.51.58      |     2 |
| 96.131.46.243   |     2 |
| 111.117.227.125 |     2 |
| 1.171.214.26    |     2 |
+-----------------+-------+
5 rows in set (0.00 sec)
```

Total page views / Category wise page views / unique page views

```
INSERT OVERWRITE DIRECTORY '/user/devendrashrivardhankar1451/project/hive/results/5/1'
row format delimited fields terminated by ','
select count(page_url) from weblog_analysis;

create table dev_result51
(
    total_page_view int
);

sqoop export --connect jdbc:mysql://cxln2.c.thelab-240901.internal/sqoopex  --table dev_result51  --export-
dir /user/devendrashrivardhankar1451/project/hive/results/5/1  --username sqoopuser --password ****** --fields-terminated-by ','
```

```
mysql> select * from dev_result51;
+-----------------+
| total_page_view |
+-----------------+
|         1000000 |
+-----------------+
1 row in set (0.00 sec)
```

```
INSERT OVERWRITE DIRECTORY '/user/devendrashrivardhankar1451/project/hive/results/5/2'
row format delimited fields terminated by ','
select page_url, count(*) from weblog_analysis group by page_url;

create table dev_result52
```

```
(
    page_url varchar(300),
    total int
);

sqoop export --connect jdbc:mysql://cxln2.c.thelab-240901.internal/sqoopex  --
table dev_result52  --export-
dir /user/devendrashrivardhankar1451/project/hive/results/5/2  --
username sqoopuser --password ****** --fields-terminated-by ','
```

```
mysql> select * from dev_result52;
+-----------------------------------------------------------------------------+-------+
| page_url                                                                    | total |
+-----------------------------------------------------------------------------+-------+
| a-o-smith-hse-sas-025-electric-instant-water-heater-white.html              |  7138 |
| ao-smith-ewsh-15-ltr-water-heater.html                                      |  7097 |
| ao-smith-hse-sbs-6-ltr-water-heater.html                                    |  6887 |
| apple-ipad-air-slate-16-gb-8155.html                                        |  6986 |
| apple-ipad-mini-slate-64-gb.html                                            |  6936 |
| apple-iphone-5s-ios-smart-phone-16-gb-black.html                            |  7019 |
| apple-ipod-shuffle-md775hn-a-mp3-player-blue.html                           |  7047 |
| apple-ipod-shuffle-md776hn-a-mp3-player-green.html                          |  7090 |
| apple-ipod-touch-mc903hn-a-mp3-player-pink.html                             |  6915 |
| apple-pro-me293-desktop-replacement-15-4-inch-39-cm.html                    |  7169 |
| asus-nexus-7-slate-7-inch-17-78-cm-tablet-32-gb.html                        |  7011 |
| bajaj-majesty-mx-3-steam-iron.html                                          |  6954 |
| canon-eos-600d-dslr-camera-black.html                                       |  7075 |
| canon-eos-60d-dslr.html                                                     |  7001 |
| canon-eos-7d-dslr-camera-black-with-ef-s18-135is-lens.html                  |  7092 |
| crompton-greaves-solarium-1-ltr-water-heater.html                           |  7163 |
| gn-netcom-jabra-solemate-mini-100-97300000-60-mini-speaker-blue.html        |  7090 |
```

```
INSERT OVERWRITE DIRECTORY '/user/devendrashrivardhankar1451/project/hive/resu
lts/5/3'
row format delimited fields terminated by ','
select distinct(page_url) from weblog_analysis;

create table dev_result53
(
    dist_page_url varchar(300)
);

sqoop export --connect jdbc:mysql://cxln2.c.thelab-240901.internal/sqoopex  --
table dev_result53  --export-
dir /user/devendrashrivardhankar1451/project/hive/results/5/3  --
username sqoopuser --password ****** --fields-terminated-by ','
```

```
mysql> select * from dev_result53;
+-------------------------------------------------------------------------------+
| dist_page_url                                                                 |
+-------------------------------------------------------------------------------+
| a-o-smith-hse-sas-025-electric-instant-water-heater-white.html                |
| ao-smith-ewsh-15-ltr-water-heater.html                                        |
| ao-smith-hse-sbs-6-ltr-water-heater.html                                      |
| apple-ipad-air-slate-16-gb-8155.html                                          |
| apple-ipad-mini-slate-64-gb.html                                              |
| apple-iphone-5s-ios-smart-phone-16-gb-black.html                              |
| apple-ipod-shuffle-md775hn-a-mp3-player-blue.html                             |
| apple-ipod-shuffle-md776hn-a-mp3-player-green.html                            |
| apple-ipod-touch-mc903hn-a-mp3-player-pink.html                               |
| apple-pro-me293-desktop-replacement-15-4-inch-39-cm.html                      |
| asus-nexus-7-slate-7-inch-17-78-cm-tablet-32-gb.html                          |
| bajaj-majesty-mx-3-steam-iron.html                                            |
| canon-eos-600d-dslr-camera-black.html                                         |
| canon-eos-60d-dslr.html                                                       |
| canon-eos-7d-dslr-camera-black-with-ef-s18-135is-lens.html                    |
| crompton-greaves-solarium-1-ltr-water-heater.html                             |
| gn-netcom-jabra-solemate-mini-100-97300000-60-mini-speaker-blue.html          |
```

Count of status code = 200 / 404 / 400 / 500

```
INSERT OVERWRITE DIRECTORY '/user/devendrashrivardhankar1451/project/hive/resu
lts/6'
row format delimited fields terminated by ','
select status_code, count(*) from weblog_analysis group by status_code;

create table dev_result6
(
    status_code int,
    total int
);

sqoop export --connect jdbc:mysql://cxln2.c.thelab-240901.internal/sqoopex  --
table dev_result6  --export-
dir /user/devendrashrivardhankar1451/project/hive/results/6  --
username sqoopuser --password ****** --fields-terminated-by ','
```

```
mysql> select * from dev_result6;
+-------------+---------+
| status_code | total   |
+-------------+---------+
|         200 | 1000000 |
+-------------+---------+
1 row in set (0.00 sec)
```