韓國IT서비스學會誌
第13卷 第4號
2014年 12月, pp.231-243

Journal of Information Technology Services
http://dx.doi.org/10.9716/KITS.2014.13.4.231

효과적인 웹 사용자의 패턴 분석을 위한 하둡 시스템의 웹 로그 분석 방안

이병주* · 권정숙** · 고기철*** · 최용락****

A Method for Analyzing Web Log of the Hadoop System for Analyzing a Effective Pattern of Web Users

Byungju Lee* · Jungsook Kwon** · Gicheol Go*** · Yonglak Choi****

■ Abstract ■

Of the various data that corporations can approach, web log data are important data that correspond to data analysis to implement customer relations management strategies. As the volume of approachable data has increased exponentially due to the Internet and popularization of smart phone, web log data have also increased a lot. As a result, it has become difficult to expand storage to process large amounts of web logs data flexibly and extremely hard to implement a system capable of categorizing, analyzing, and processing web log data accumulated over a long period of time.

This study thus set out to apply Hadoop, a distributed processing system that had recently come into the spotlight for its capacity of processing large volumes of data, and propose an efficient analysis plan for large amounts of web log. The study checked the forms of web log by the effective web log collection methods and the web log levels by using Hadoop and proposed analysis techniques and Hadoop organization designs accordingly. The present study resolved the difficulty with processing large amounts of web log data and proposed the activity patterns of users through web log analysis, thus demonstrating its advantages as a new means of marketing.

Keyword : Hadoop, Web Log

Submitted : July 25, 2014 1st Revision : December 5, 2014 Accepted : December 10, 2014

* 숭실대학교 정보과학대학원 석사, 주저자
** 숭실대학교 SW특성화대학원 석사, 공동저자
*** 숭실대학교 일반대학원 박사, 공동저자
**** 숭실대학교 SW특성화대학원 교수, 교신저자

1. 서 론

고객관계관리(CRM)란 고객의 행동 패턴에 대한 분석을 통해 기업 경영의 효과를 높이기 위한 일련의 과정을 의미한다. 성공적인 고객관계관리는 고객의 특성에 대한 정확한 이해에서 출발한다(Kang et al., 2006). 초기에는 고객의 욕구변화와 행동패턴에 주목하여 데이터를 분석 가능한 형태로 변환하여 분석하는 데이터 마이닝 기법을 많이 사용하였고, 최근에는 기업이 보유하고 있는 고객 데이터, 상품 데이터 혹은 각종 활동을 통한 데이터의 분석을 통해 새로운 경향과 규칙을 발견하여 상품 개발 및 비즈니스 의사결정에 반영하고 이를 마케팅에 적극적으로 활용하고 있다.

IT 기술의 발전으로 대용량 데이터베이스, 데이터 웨어하우스 구축이 가능해짐에 따라 축적하고 사용 가능한 데이터의 종류와 양은 기하급수적으로 증가하고 있다. 웹 로그 데이터는 고객관계관리 전략 수행 및 마케팅을 위한 데이터 분석에 부합하는 중요한 데이터이다(Oh, 2011). 웹 로그 데이터란 고객이 웹 사이트에 접속하여 실행했던 기록으로서 고객이 인터넷에 접속해 서비스를 사용하는 일련의 과정과 행위들을 파악할 수 있으며, 이를 기반으로 향후 고객의 새로운 서비스 요청 시 좀 더 효과적인 의사결정을 도울 수 있다(Kim and Min, 2001).

온라인 비즈니스와 오프라인 비즈니스의 차이 중 한 가지는 정보의 수집과 활용이다. 오프라인 마켓은 고객의 행동과 유형, 성향을 파악하기 어렵지만, 인터넷상의 온라인 마켓에서는 이 모든 정보를 수집하고 마케팅에 활용할 수 있다(Jung and Lee, 2003). 이러한 온라인 비즈니스만의 장점을 가능케 하는 것이 바로 웹 로그 분석이다. 하지만 인터넷과 스마트폰의 대중화로 접근 가능한 데이터의 양이 증가함에 따라 웹 로그 데이터 또한 증가하였고, 대용량 웹 로그 데이터 처리를 위한 웹 로그 데이터 분류, 분석처리 기능을 제공하는 시스템을 한정된 자원을 가지고 기존의 기술로 구현하기가 매우 어려워졌다. 그리고 사용 가능한 데이터가 지속해서 축적되어 있음에도 불구하고 많은 양과 복잡성으로 인해 데이터의 효율적인 사용이 더욱 어려워졌다.

이에 대용량의 웹 로그 데이터 처리를 위해 오픈소스 기반의 대용량 데이터 플랫폼으로 분산 환경에 최적화되어있는 대용량 실시간 프로세싱 프레임워크인 하둡(Hadoop)을 적용하여 웹 로그 분석을 해보고자 한다.

본 논문에서는 효과적인 웹 로그 수집 방법과 웹 로그 레벨별로 발생하는 웹 로그의 형태를 확인하고 이에 맞는 분석 기법 및 하둡의 구성 설계를 제안하고자 한다.

2. 관련 연구

2.1 웹 로그 데이터 분석

웹 로그 파일이란 웹 서버를 통해 이루어지는 모든 작업에 대한 기록이다. 웹 사이트를 방문한다는 것은 브라우저를 통해 웹 서버에 필요한 정보를 요청하는 것이고, 이때 방문정보, 활동내용, 활동시간 등은 웹 서버에 미리 저장해 놓은 위치에 데이터로 남게 된다. 웹 로그 데이터를 분석하는 궁극적인 목적은 고객들의 관심이 어디에 있는지를 파악하고 고객들의 반응을 예측하여 회사에 비즈니스에 전략적으로 활용할 수 있도록 하는 데 있다(Kim and Min, 2001).

사용자가 웹 사이트를 이용하면서 남긴 로그 데이터를 기반으로 다양한 활용을 할 수 있다.

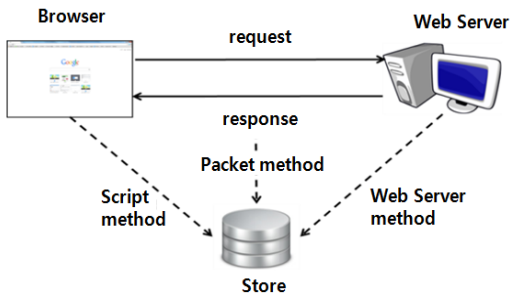
첫째 : 개별 고객들의 성향을 파악하여 고객들이 원하는 맞춤 서비스를 제공할 수 있다.

둘째 : 사이트 정보설계를 경제적으로 할 수 있다.

셋째 : 특정 대상에 대한 집중적인 마케팅과 광고 전략 수립이 가능하다.

넷째 : 최적의 환경에서 사용자들이 사이트를 탐색하고 방문하도록 서버 및 회선 등의 기술적 자원을 계획하거나 수립할 수 있다(Kim, 2000).

웹 로그 파일은 사용자가 처음 사이트를 방문하면 웹 서버에 자동으로 자신의 로그파일이 웹 서버에 저장되어 로그파일이 생성되게 된다. 이러한 웹 로그 파일은 액세스 로그, 레퍼럴 로그, 에이전트 로그, 에러 로그 파일과 같이 크게 4가지로 분류할 수 있다(Lee, 2004).



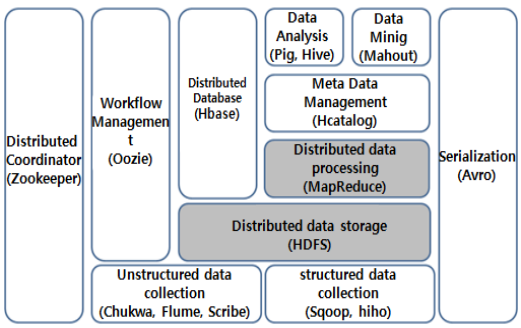
〈Figure 1〉 Web Log Extraction Method

일반적으로 많이 사용되고 있는 웹 로그 추출 방법은 웹 서버 방식, 스크립트 방식, 패킷 방식이 있으며 때에 따라서 2가지 이상의 추출 방법을 혼합시 사용되기도 한다. <Figure 1>은 이를 도식화한 그림이다(Jang, 2002).

기존에는 대용량 데이터를 분석하기 위해 싱글 머신 멀티코어에 최적화된 고사양의 서버를 이용해서 DW, DM을 구축하고, 이러한 장비에 최적화된 고비용의 소프트웨어를 사용했다. 그러나 최근에는 인터넷을 사용하는 고객들의 사용로그와 트랜잭션 로그를 기반으로 하는 서비스나 플랫폼을 구축하고자 할 시 대용량의 비정형 데이터와 다양한 미디어 정보를 실시간으로 수집해서 분석해야 하므로 기존 시스템과 소프트웨어 아키텍처로는 비용과 기술에 한계가 있다.

2.2 하둡

하둡은 구글, 야후, 페이스북, 아마존 등 인터넷 서비스를 자체 운영해온 기업들이 빅데이터 처리를 위해 연구・적용해온 오픈 소스 기술이다. 하둡은 대량의 자료를 처리할 수 있는 대규모 컴퓨터 클러스터에서 동작하며 분산 애플리케이션을 지원하는 오픈 자바 소프트웨어 프레임워크이다. 이러한 하둡의 구성요소는 분산 파일 시스템인 HDFS와 분산 데이터베이스 모델인 Hbase 그리고 대용량 데이터 처리를 위한 맵리듀스이며 <Figure 2>는 하둡 시스템의 주요 구성요소를 보여준다.



〈Figure 2〉 Hadoop Framework(Jung, 2012)

HDFS와 Hbase는 저장된 거대한 데이터 셋을 간편하게 분산 처리하는 Java 기반의 맵리듀스 프레임워크를 제공한다. 이외에도 하둡을 기반으로 다수의 오픈소스 분산처리 프로젝트가 존재한다(Jung, 2012; Tom, 2009; Thusoo et al., 1996).

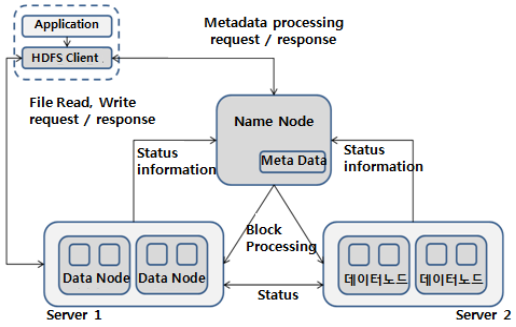
하둡 프로젝트의 최대 후원자인 야후에서는 4,000여 대의 서버로 단일 HDFS 클러스터를 구성하여 페타바이트 이상의 데이터를 저장, 관리하고 있다. 블록들이 클러스터를 구성하는 전체 노드와 전체 디스크에 고르게 분산되어 저장되기 때문에 디스크 IO 성능이 중요한 맵리듀스 같은 데이터 중심에 병렬처리 연산에 최적화된 파일 시스템이라고 할 수 있다. 저장되어있는 각 블록은 데이터 유실의 위험이나 사람들이 많이 접속할 때의 부하 처리를 위해 각 노드 클러스터에 복사된다. HDFS는 다음과 같은 특징을 목표로 한다(Jung, 2012; Apache Hadoop, 2013).

- 장애복구
- 스트리밍 방식의 데이터 접근

- 대용량 데이터 저장
- 데이터 무결성

HDFS는 마스터(Master)-슬레이브(Slave) 아키텍처로 구성된다. 즉, 마스터 역할을 하는 네임노드(NameNode) 서버가 한 대, 슬레이브 역할을 하는 데이터노드(DataNode) 서버가 여러 대로 구성된다.

네임노드는 HDFS의 모든 메타 데이터를 관리하고, 클라이언트가 HDFS에 저장된 파일에 접근할 수 있게 해준다. HDFS에 저장할 때 블록으로 나뉜 데이터는 여러 대의 데이터노드에 분산 저장된다. 사용자가 구현한 애플리케이션은 HDFS에 파일을 저장하거나, 저장된 파일을 읽기 위해 HDFS 클라이언트를 사용하며, 클라이언트는 API 형태로 사용자에게 제공된다. <Figure 3>은 HDFS 도식화하여 나타냈다.



〈Figure 3〉 HDFS Structure(Jung, 2012)

구글에 의해 고안된 맵리듀스는 함수형 프로그래밍 언어인 LISP(List Processor)를 모델로 하여 맵과 리듀스라는 2개의 함수를 이용한 대용량 데이터를 쉽고 빠르게 처리할 수 있는 방법론이다.

맵리듀스는 작업 수행 시 데이터 지역성을 보장하여 네트워크 트래픽 발생을 최소화시켜주고 데이터가 있는 곳에서 프로세스가 실행될 수 있도록 스케줄링을 한다. 그리고 노드의 수행 도중 고립의 장애 발생 시 자동 복구를 위해 데이터 복제본이 있는 다른 서버(데이터 노드)에서 작업을 재 시작하여 중단없이 처리를 계속한다. 그리고 노드의 확장이 유연하므로 성능향상을 기대할 수 있다. 맵리듀스 시스템은 클라이언트, 잡트래커, 태스크트래커로 구성된다(Tom, 2009; Dean, 2008; Jung, 2012).

클라이언트가 하둡으로 실행을 요청하는 맵리듀스 프로그램은 잡(Job)이라는 하나의 작업 단위로 관리된다. 잡트래커는 하둡 클러스터에 등록된 전체 잡의 스케줄링을 관리하고 모니터링한다. 전체 하둡 클러스터에서 하나의 잡트래커가 실행되며 하둡의 네임노드 서버에서 실행된다.

태스크트래커는 사용자가 설정한 맵리듀스 프로그램을 실행하며, 하둡의 데이터노드에서 실행되는 데몬이다. 태스크트래커는 잡트래커의 작업을 요청받고, 잡트래커가 요청한 맵과 리듀스 개수만큼 맵 태스크(Map Task)와 리듀스 태스크(Reduce Task)를 생성한다.

맵 태스크와 리듀스 태스크란 사용자가 설정한 맵과 리듀스 프로그램이다. 맵 태스크와 리듀스 태스크가 생성되면 새로운 실행환경을 구동해 맵 태스크와 리듀스 태스크를 실행한다.

3. 제안하는 웹 로그 분석 방법

3.1 기능분석

제안하는 웹 로그 분석은 대용량 데이터를 효율적으로 처리할 수 있어야 하며, 데이터 무결성 또한 보장되어야 한다. 이에 필요한 기능적인 요소들은 아래와 같다.

첫째 : 확장 가능한 대용량 데이터 처리
웹 로그 분석 시 분석 로그의 단위와 크기가 큰 규모의 웹 로그 데이터를 효율적으로 처리할 수 있어야 하며 데이터 규모가 확장됨에 따라서 이를 유연하게 처리하는 방안도 고려되어야 한다.

둘째 : 장애 발생에 따른 시스템의 연속성 보장
장애는 IT 기반 구조에 영향을 주는 예상치 못한 사건으로 정의되며, 시스템 장애의 원인은 무수히

많이 존재한다. 따라서 웹 로그 분석 시에는 다양한 장애 상황에 대비하여 웹 로그 분석처리의 연속성을 보장할 수 있는 설계의 필요성이 고려되어야 한다.

셋째 : 사용자 인터페이스
웹 로그 분석을 통해 산출되는 결과는 비즈니스 환경에 따라서 다양한 형태로 제공되어야 한다. 웹 로그 분석을 통해 웹 사용자의 다양한 행동패턴에 대한 분석 자료는 결국 또 다른 비즈니스 사업에 대한 기반 자료가 될 수 있다. 이를 위해서 분석 데이터는 각각의 비즈니스 단위에 적합한 형태로 제공되어야 한다.

3.2 웹 로그 분석을 위한 하둡 설계

제안하는 시스템의 포괄적인 흐름 구성은 <Figure 4>와 같이 이루어진다.

〈Figure 4〉 System Flow

위의 5단계 흐름 구성의 특징은 아래와 같다.

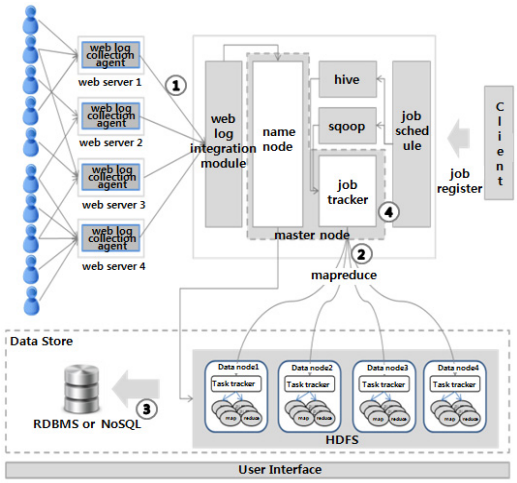
첫째 : 웹 로그 발생 단계는 사용자가 웹에 접근하여 접속 종료 시까지 발생하는 분류와 검증 없이 무작위로 발생하는 일련의 사용자 활동 기록이다.

둘째 : 수집단계는 각 웹 로그를 수집할 서버에 수집 모듈이 일정 시간 단위로 발생한 웹 로그파일을 마스터 서버로 전송하는 단계이다.

셋째 : 적재 단계는 수집모듈이 전송한 웹 로그를 HDFS 저장소에 일정 시간 단위로 디렉터리를 생성하여 저장하는 단계이다.

넷째 : 분석 단계는 HDFS에 적재된 로그 데이터를 맵리듀스 기반으로 해석을 수행하는 단계이다.

다섯째 : 분석 결과 저장 단계는 웹 로그 데이터 분석이 끝나면 생성된 결과를 DBMS에 저장하는 단계이다.



〈Figure 5〉 Web Log Analyzing System Structure

<Figure 5>처럼 웹 로그 분석 제안 시스템은 기본적으로 하둡의 HDFS와 맵리듀스를 활용하였다. 또한, 다양한 에코 시스템을 활용함으로써 효과적인 웹 로그 분석을 수행할 수 있게 된다. 각 시스템의 단위 기능은 웹 로그 수집/적재, 웹 로그 분석, 웹 로그 분석결과 처리, 분석 스케줄 기능으로 분류할 수 있고, 웹 로그 통합 모듈의 상세 내용은 아래와 같다.

첫째 : 웹 로그 수집/적재 모듈은 웹 로그를 수집할 서버에 플럼 Agent에 의해 일정 시간 단위로 발생한 웹 로그 파일들을 에브로 프로토콜을 사용하여 하둡 마스터 서버 플럼 Collector로 전송하게 된다. 플럼 Collector는 수신받은 로그를 HDFS 저장소에 디렉터리를 생성하여 저장한다. 이 작업은 일정 시간 단위로 반복하여 수행되면서 각 서버에 있는 모든 웹 로그 파일을 한곳으로 모아주는 역할을 한다. 본 논문에서는 수집시간을 10분 간격으로 설정하였다.

둘째 : 웹 로그 분석 모듈은 HDFS에 적재된 웹 로

그 데이터를 맵리듀스 기반으로 해석하는 역할을 수행한다. 주기적인 분석을 위해 맵리듀스가 언제 몇 분 단위로 잡을 실행 할지에 대한 정보를 분석 스케쥴 모듈에 스케줄 정보를 등록하고 등록된 스케줄 정보에 따라 하둡의 잡트래커에게 실행명령을 내리게 되고 명령을 받은 잡트래커는 실제 병렬로 분석작업을 처리할 각 테스크노드들에게 작업명령을 하달한다.

셋째 : 웹 로그 분석 결과 처리 모듈은 각 태스크 노드들은 노드에서 분석할 로그파일을 가지고 분석을 수행하게 되며 데이터 분석이 끝나면 결과 파일을 HDFS 저장소에 생성하게 된다. 생성 된 결과 데이터는 RDBMS 또는 NoSQL 기반의 DBMS에 저장함으로써 한 개의 잡이 실행을 마친다.

넷째 : 분석 스케줄 모듈은 주기적으로 웹 로그 적재 모듈, 웹 로그 분석 모듈을 주기적으로 호출해주는 역할을 수행한다.

3.3 웹 로그 분석 시스템 모듈 구성

3.3.1 웹 로그 수집/적재 모듈 구성

- 웹 로그 수집 모듈

본 논문에서 설계된 웹 로그 수집 모듈은 플럼 모듈을 사용하여 구성하였다. 플럼은 분산 환경에 만들어진 대용량의 웹 로그 데이터를 효율적으로 옮길 수 있는 로그 수집기이다. 웹 로그의 수집과 저장은 설정파일에 의해 이루어지며, 설정파일을 주기적으로 확인하여 설정 변경 여부 확인하고 변경된 설정정보를 동적으로 플럼 모듈에 적용할 수 있다. <Figure 6>은 플럼 Collector 설정파일 화면이다.

```
# agent 설정
collector_agent.sources = avro-source
collector_agent.sinks = hadoopSink
collector_agent.channels = channel1

# channel 설정
collector_agent.channels.channel1.type = memory
collector_agent.channels.channel1.capacity = 1000
collector_agent.channels.channel1.transactionCapactiy = 100

# sources 설정
collector_agent.sources.avro-source.type = avro
collector_agent.sources.avro-source.bind = 192.168.159.100
collector_agent.sources.avro-source.port = 8994

# sinks 설정
collector_agent.sinks.hadoopSink.type = hdfs
collector_agent.sinks.hadoopSink.hdfs.path = flume/weblog
collector_agent.sinks.hadoopSink.hdfs.fileType = DataStream
#collector_agent.sinks.hadoopSink.hdfs.batchSize = 2000
#collector_agent.sinks.hadoopSink.hdfs.writeFormat = Text
#collector_agent.sinks.hadoopSink.hdfs.rollSize = 1000000
#collector_agent.sinks.hadoopSink.hdfs.rollInterval = 3600
#collector_agent.sinks.hadoopSink.hdfs.rollCount = 1024

# channel에 sources와 sinks 바인딩
collector_agent.sources.avro-source.channels = channel1
collector_agent.sinks.hadoopSink.channel = channel1
```

〈Figure 6〉 Flume Collector Module Setting

해당 정보의 source 설정의 bind 정보의 IP 주소는 하둡 서버의 주소이다. 또한, sinks 설정을 보면 sink의 type이 hdfs이고 “flume/weblog”에 저장되는 것을 볼 수 있다. 그 밖의 저장방법에 대한 옵션을 기반으로 플럼은 하둡 서버의 HDFS에 웹 로그 정보를 보내주게 된다(Apache Flume, 2013).

- 웹 로그 적재 모듈

웹 로그 적재 모듈은 HDFS에 저장한 웹 로그 데이터를 하이브의 HiveQL 명령어를 이용하여 통합하는 역할을 수행한다. 해당 모듈은 웹 로그 수집기로부터 수집된 웹 로그 데이터를 그대로 저장할 source 테이블과 웹 로그 데이터의 엘리먼트 값들을 추출하여 별도로 저장할 web log 테이블을 생

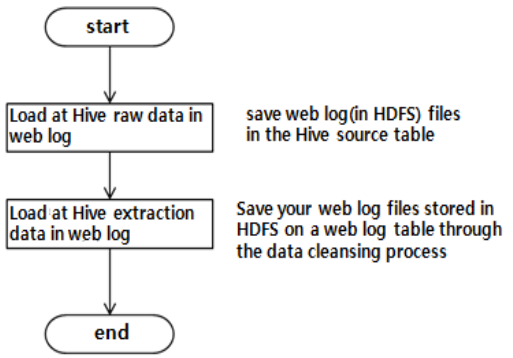
〈Table 1〉 Source Table Structure

Field Name	TYPE	Description
WebLogFile	String	web log Raw Data

〈Table 2〉 Web Log Table Structure

NO	Field Name	TYPE	Description
1	Host	String	Domain Name or IP Address
2	User	String	It there is no user name, write “-”
3	Time	String	Date and time of access
4	Request	String	Transfer Protocol
5	Status	String	Connection and data movement state
6	Size	String	Transmitted data size
7	Agent	String	access environment information of the users

성하며, 생성되는 테이블과 흐름도는 아래와 같다.

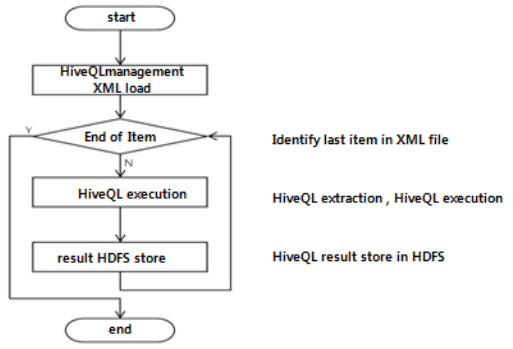


〈Figure 7〉 Web Log Stack Module Flowchart

3.3.2 웹 분석 모듈 구성

웹 로그 분석 모듈은 web log 테이블에 저장된 웹 로그 데이터를 기반으로 하이브의 내장 함수와 HiveQL 명령어를 이용하여 분석하는 역할을 수행한다. 웹 로그 분석은 비즈니스 환경에 따라 다양한 분석 요건이 발생할 수 있으므로, 본 시스템에서는 분석 요건인 HiveQL를 XML로 관리하여 요건 발생 시 쉽게 추가할 수 있는 기능을 제공한다.

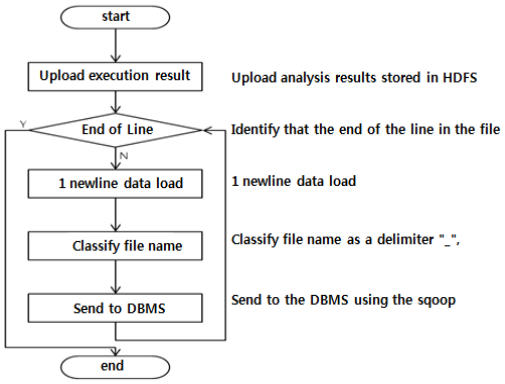
<Figure 8>은 웹 로그 분석 모듈 흐름도이다.



〈Figure 8〉 Web Log Analyzing Module Flowchart

3.3.3 웹 로그 분석 결과 처리 모듈 구성

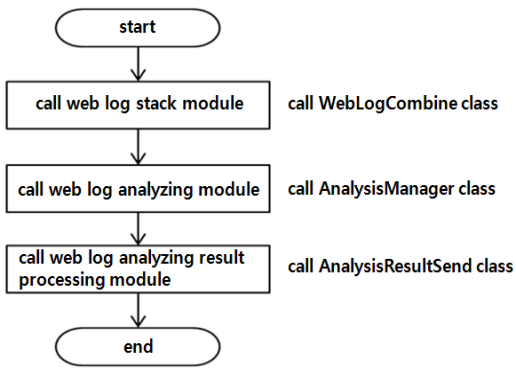
본 논문에서 설계된 웹 로그 분석 결과 처리 모듈은 AnalysisResultSend 클래스로 정의하였으며, AnalysisResultSend 클래스는 HDFS에 저장된 결과 데이터를 스쿱의 명령을 이용하여 DBMS로 전송하는 역할을 담당하게 된다. <Figure 9>는 웹 로그 분석 결과 처리 모듈 흐름도이다.



〈Figure 9〉 Web Log Analyzing Result Process Module Flowchart

3.3.4 웹 로그 분석 스케줄 모듈 구성

본 논문에서 설계된 웹 로그 분석 스케줄 모듈은 웹 로그 적재, 분석, 결과처리 모듈을 주기적으로 실행시키는 역할을 담당한다. 웹 로그 분석 스케줄 모듈은 각각의 모듈을 실행시킬 수 있는 클래스가 필요하며, 본 논문에서는 이를 AnalysisSchedule 라고 명명하였다. 웹 로그 분석 스케줄 모듈은 10분마다 실행되도록 설정하였으며, <Figure 10>은 웹 로그 분석 스케줄 모듈 흐름도이다.



〈Figure 10〉 Web Log Analyzing Schedule Module Flowchart

4. 웹 로그 분석 수행

4.1 웹 로그 분석 데이터

분석 수행을 위해서 사용되는 웹 로그는 특정 기업에서 사용하는 사내 웹 사이트를 통해 축적된 웹 로그를 받아 진행하며, 해당 웹 로그의 양은 하루 평균 150MB 크기이고 6개월간의 수집한 데이터를 활용하였다. 또한, 실시간으로 발생하고 있는 웹 로그의 수집은 로그 수집기 모듈을 활용하여 수집하였다.

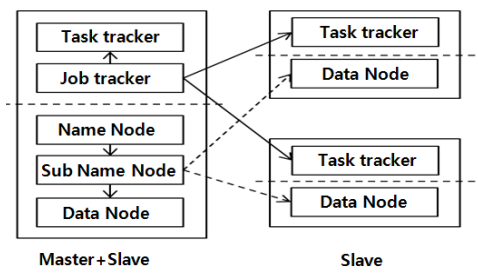
4.2 웹 로그 분석 환경 구축

하둡 프레임워크 구성을 위해서 본 논문에서는 3개의 서버를 사용하였다. 각 서버의 사양은 <Table 3>과 같으며, 3대의 사양은 모두 같다. 1대 서버에는 마스터 노드와 슬레이브 노드가 함께 구동되며 나머지 2대의 서버에서 슬레이브 노드를 구축하였다.

〈Table 3〉 Hadoop Server Specifications

Category	Specifications and Versions
CPU	2.13 GHz
RAM	1GB
HDD	100GB
OS	Linux, Ubuntu 11.04
HADOOP	0.20.2

구축된 하둡 서버 구조는 <Figure 11>과 같다.



〈Figure 11〉 Build Hadoop Server Structure

4.3 웹 로그 분석

분석 방법은 아래 <Table 4>와 같이 접속 분석, 페이지 분석, 방문자 분석, 시스템 환경 분석과 같이 4가지 영역별로 분석할 수 있다.

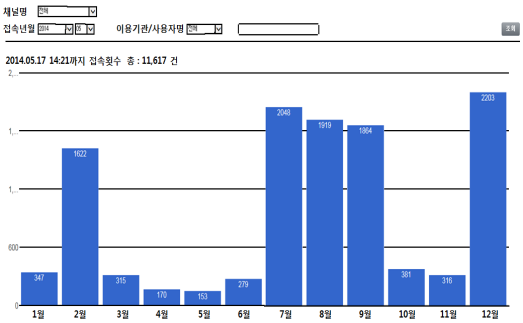
〈Table 4〉 Web Log Analyzing Type

Category	Analyzing Type
Access Analyzing	Monthly/Daily/Hourly Access Trend Analyzing
Page Analyzing	Top-up Page Analyzing
Visitor Analyzing	Visit Frequency Analyzing
System Environment Analyzing	Access Browser Analyzing
	Access Operating System Analyzing

- 접속 분석

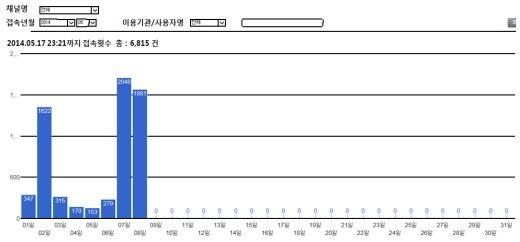
사이트를 방문한 사용자의 접속 유형 분석을 통하여 접속빈도, 집중 접속시간 등의 사이트 관리를 위한 정보 추출을 목적으로 한다. 이러한 분석은 월별, 일별, 시간대별 분석이 가능하며 이러한 분석결과는 분석데이터의 신뢰성을 높이며, 정밀한 통계결과를 제공할 수 있다.

<Figure 12>는 월별로 접속한 사용자 수의 추이를 분석한 그래프이다.



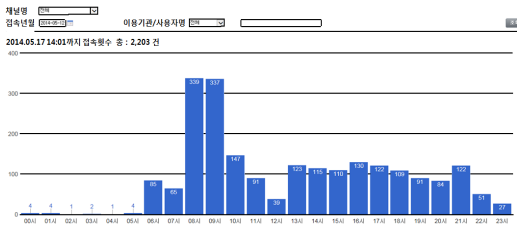
〈Figure 12〉 Monthly Access Trend Graph

<Figure 13>은 일별로 접속한 사용자 수의 추이를 분석한 그래프이다.



〈Figure 13〉 Daily Access Trend Graph

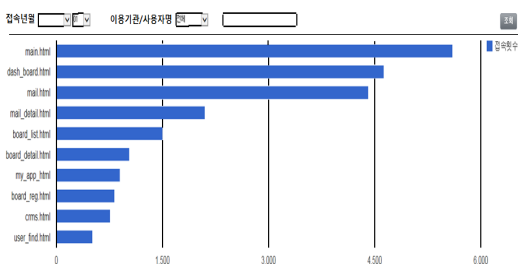
<Figure 14>는 시간대별로 접속한 사용자 수의 추이를 분석한 그래프이다.



〈Figure 14〉 Hourly Access Trend Graph

- 페이지 분석

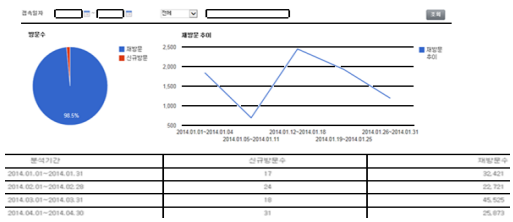
사이트를 방문한 사용자가 어떤 페이지에 많이 접속했는지에 대한 분석을 통하여, 해당 페이지에 대한 마케팅을 고려해볼 수 있는 정보 추출을 목적으로 한다. <Figure 15>는 사이트에 방문한 방문자들이 많이 조회한 페이지 분석 그래프이다.



〈Figure 15〉 Top-up Page Graph

- 방문자 분석

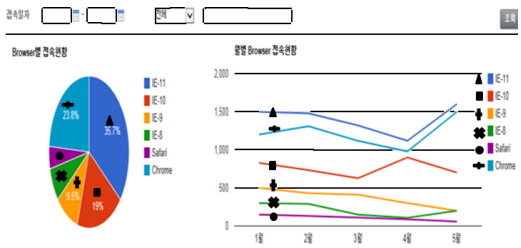
사이트에 유입한 방문자의 방문빈도 분석을 통해 사이트의 콘텐츠나 상품의 구성이 충실한지에 대한 평가를 가늠할 수 있는 지표를 목적으로 한다. <Figure 16>은 재방문과 신규방문에 대한 비교 분석한 그래프이다.



〈Figure 16〉 Visit Frequency Analyzing Graph

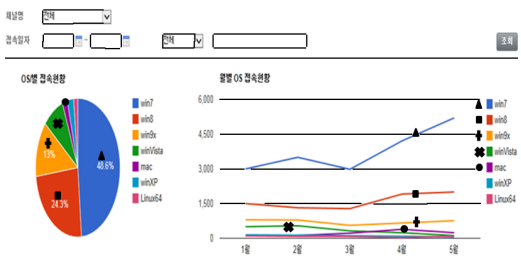
- 시스템 환경 분석

사이트에 방문한 방문자들이 사용하는 접속 브라우저와 운영체제를 분석하여 방문객의 환경을 이해할 수 있으며, 방문객의 환경에 맞도록 사이트를 최적화하는 데 목적이 있다. <Figure 17>은 방문자들이 사용하는 웹 브라우저 종류를 분석한 그래프이다.



〈Figure 17〉 Access Browser Analyzing Graph

<Figure 18>은 방문자들이 사용하는 운영체제 종류를 분석한 그래프이다.



〈Figure 18〉 Access Operating System Analyzing Graph

4.4 시스템 평가

본 장에서는 상업적으로 운영되고 있는 웹 로그 시스템과 비교·분석하여 하둡 웹 로그 분석 시스템을 평가하고자 한다.

<Table 5>에서 제시한 비교항목으로 하둡 웹 로그 분석 시스템과 일반적으로 사용하는 웹 로그 시스템을 비교하였다. 제시한 비교항목은 객관적인 평가를 할 수 있는 필수 요소만 평가하였다.

〈Table 5〉 Comparison with Other Web Log System

Category / Compare	Weg log File Analyzing System	Page-embedding System	Hadoop web log Analyzing System
web log Extraction Script Operations	Not Required	Required	Not Required
Disk Capacity	Plenty	Usually	Plenty
Marketing Oriented Analysing	Impossibility	Available	Available
Real-time Data Collection	Not Supported	Supports	Periodic Collection Available
Links with Existing DBMS	Impossibility	Available	Available
Data Collection Methods	Recorded in the Disk of the Server	Remote Collected by the Script	Recorded in the Disk of the Server

하둡 웹 로그 분석 시스템을 타 시스템에서 제안한 내용과 비교하여 서술하면 다음과 같이 평가할 수 있다.

첫째 : 처리구성요소의 단순화를 들 수 있다.
일반적으로 상업적인 용도로 사용하며 다양한 기능을 제공하는 로그 시스템은 마케팅중심의 분석을 위해서 페이지별로 스크립트를 삽입하는 페이지 인베딩 방식을 사용하고 있다. 하지만 이러한 방식은 많은 웹 페이지의 수정이 필요하므로 적용하기가 쉽지 않다. 반면에 본 논문에서 제안한 시스템은 웹 서버가 기본적으로 제공하는 웹 로그를 활용하여 같은 효과를 내므로 별도로 페이지에 삽입되는 스크립트는 필요가 없다.

둘째 : 시스템의 성능유지에 향상을 준다.
웹 로그 데이터는 일반적으로 그 양이 많아 자칫 서버의 전체적인 성능저하를 가져올 수도 있다. 본 논문에서 제안하는 시스템에서는 대용량 데이터 처리에 적합한 하둡을 적용하여 대용량의 웹 로그 데이터의 수집 및 분석에 대한 문제점을 해결하였다.

셋째 : 다양한 통계적 결과를 얻을 수 있다.
일반적인 웹 로그 분석 시스템은 시스템적 퍼포먼스 측정(방문 사용자 수, 페이지뷰, 히트 수 등)과 같은 기본적인 데이터를 관리하는 데 반해, 제안 시스템은 시스템적 퍼포먼스 측정 데이터뿐만 아니라 방문자의 특성과 행동양식과 같은 마케팅에 필요한 측정 데이터를 얻을 수 있다.

넷째 : 추출된 로그데이터와 기존의 DBMS 데이터 간의 연계가 가능하므로 웹 로그 분석 시 사용자, 페이지화면 등과 같은 유용한 정보를 얻을 수 있어 실질적인 웹 로그 분석이 가능하다.

다섯째 : 유지보수를 위한 별도의 노력이 필요 없다.
신규 페이지가 생성될 경우 기존의 웹 로그 시스템은 해당 페이지의 로그추출을 위해서 별도의 스크립트 추가 작업을 해야 한다. 하지만, 제안 시스템에서는 HiveQL관리 XML에 새로운 요건에 정보를 추가하면 된다.

여섯째 : 자료수집의 효율성을 제공한다.
웹 로그 파일과 같이 특정 파일의 내용 분석을 통한 자료수집방법이므로 해당 파일이 웹 서비스의 운영에 사용 중일 때는 실시간의 정보수집이 어렵

다는 단점이 있지만, 하둡 웹 로그 분석 시스템에서는 주기적인 수집이 가능하다는 장점이 있다.

5. 결론 및 향후 연구 과제

본 논문에서는 하둡 시스템을 이용하여 대용량의 웹 로그를 수집, 분석하여 사용자들의 행동 패턴과 성향 등을 통계적 자료로 제공할 수 있는 시스템을 설계 구현하였다. 제안하는 시스템의 검증을 위해서 운영환경을 구축하여 구현 가능성을 확인하였으며, 성능적인 면과 유지운영 측면에서 아래와 같은 결과를 얻을 수 있었다.

성능 측면에서는 웹 로그 데이터의 크기가 증가하더라도 하둡 클러스터 상에서 자동으로 분산 병렬 처리되어 웹 로그 분석 시간이 일정하게 유지되었으며, 작업 로그 데이터의 크기가 증가하더라도 전체 분석시간에 영향을 주지 않고 실행됨을 확인할 수 있었다.

유지운영 측면에서는 병렬 분산 처리방식인 하둡 시스템은 하둡 클러스터 노드가 대규모로 확장되어도 이를 유연하게 처리하기 때문에 확장성이 쉬우며, 특정 노드에 장애 발생 시 다른 노드에 복제 데이터가 존재하여 데이터의 무결성을 유지 할 수 있다. 또한, 웹 로그 수집 모듈과 분석 모듈, 스케줄 모듈을 적용하여 웹 로그 수집, 분석을 주기적으로 수행하여 기존 웹 로그 분석 시스템의 단점을 보안 할 수 있었다. 그리고 웹 로그 분석에 이용한 하이브는 확장 가능한 구조로 인해 다양한 웹 로그 분석 요건을 수용할 수 있어 요건 추가 시 유연하게 처리할 수 있다.

하지만 본 논문은 실험을 위해 사용된 웹 로그의 다양성에 대한 검증이 부족하였고 향후에는 더 다양한 패턴의 데이터와 많은 양의 자료를 수집할 필요가 있다. 또한, 하둡을 사용한 시스템은 각 노드 사이의 데이터 전달을 위한 네트워크 I/O 등이 원인으로 저용량 데이터에서는 만족할 만한 속도가 도출되지 않기 때문에 이러한 저용량 데이터 분석에 대한 성능개선 또한 향후 연구를 통해 진행할 예정이다.

본 논문에서 제안하는 시스템은 단순히 텍스트 분석뿐만 아니라 비정형 데이터인 이미지나 음원 파일 등을 분석할 수 있는 시스템으로서의 확장성 또한 향후 연구과제로 제시할 수 있다.

References

Apache Flume, http://flume.apache.org/ (Accessed May 20. 2013).

Apache Hadoop, http://wiki.apache.org/hadoop (Accessed May 20. 2013).

Dean, J., S. Ghemawat, “MapReduce : Simplified Data Processing on Large Clusters”, *Communications of the ACM*, Vol.51, No.1, 2008, 107-113.

Go, Y.D., *Design and Implementation of Web Analyzing System based on User Create Log*, Korea National Open University Graduate School Master’s Thesis, 2007.

Han, J., M. Kamber, and J. Pei, *Data Mining : Concepts and Techniques*, Second edition, ELSEVIER Inc, New York, 2006.

Hive, http://wiki.apache.org/hadoop/Hive (Accessed January 16. 2013).

Jang, N.S., *Data Mining*, Daechung Media, 1999.

Jang. Y.K., *A study on the Relationship between Customer’s Action and Customer’s Value using Log Analysis*, Graduate School of Ajou Master’s Thesis, 2002.

Jung, J.H., *Get started! Hadoop Programming*, Wikibooks, 2012.

Jung, S.K. and C.W. Lee, “Web long Data Analysis Apply to Web Contents Analysis methodology”, *HCI Conference of Korean Institute of Information Scientists and Engineers*, Vol.2, 2003, 1462-1467.

Kang, R.G., H.K. Lim, and C.Y. Jung, “Datamin-

ing technique for successful eCRM, CRM", *Journal of Korea Institute of Information and Communication Engineering*, Vol.10, No.9, 2006, 1596-1601.

Kim, B.S., "The Role of Site Stickiness and Its Antecedents in a Social Commerce Environment", *Journal of Information Technology Services*, Vol.12, No.3, 2013, 23-37.

Kim, H.T., *Internet Marketing.com*, Triangle M &B, 2000.

Kim, H.T. and O.G. Min, *Web Log Analytics*, Bibicom, 2001.

Kim, S.H. and H.S. Park, "An Empirical Study on Individual and Social Commerce Factors Impacting Shopping Value and Intention to Repurchase in Social Commerce and Moderating Effects of Perceived Security", *Journal of Information Technology Services*, Vol.12, No.2, 2013, 31-53.

Lee, J.I., K.M. Baek, J.H. Shin, and W.S. Lee, "Building Data Warehouse System for Weblog Analysis", *Conference of Information Technology Services*, Vol.2010, No.1, 291-295.

Lee, S.J., *A Case Study of Weblog Analysis-from a Path Analytic Point of View*, Graduate School of Dankook Master's Thesis, 2004.

Oh, J.H., J.H. Kim, and J.W. Kim, "A Study on the Development of Realtime Online Maketing System Using Web Log Analytics", *Journal of Society for e-Business Studies*, Vol.16, No.3, 2011, 249-261.

Thusoo, A., J. S. Sarma, N. Jain, Z. Shao, P. Chakka, N. Zhang, S. Antony, H. Liu, and R. Murthy, "Hive-a petabyte scale data warehouse using hadoop", *IEEE ICDE*, Proceedings of the 26th IEEE International Conference on Data Engineering(ICDE), 2010, 996-1005.

Tom, W., *Hadoop : The Definitive Guide*, O'Reilly Media, 2009.

About the Authors

Byungju Lee (lbjj1983@naver.com)

Byungju Lee received the M.S. degree in software engineering from soongsil University in 2014. He has been working for WebCash co.,ltd. since then. His current research interests include Distributed System, data mining, and Hadoop distributed file system.

Jungsook Kwon (pippie@daum.net)

Jungsook Kwon received degrees in Computer Science, Education, Sookmyung Women's University, February 1995. She is currently in a master course of Software, Soongsil University. She worked for LG from 1996 to 2012. Her current research interests include Advanced financial system, Project Management, data mining and Big Data.

Gicheol Ko (jeada4@naver.com)

Gicheol Ko is currently in a Doctor course of IT Policy and Management Dept., Soongsil University. He worked in IT for 20 more years. His main interests are Big Data, Data Modeling, OLAP, Information analysis, Information security.

Yonglak Choi (ylchoi58@ssu.ac.kr)

Professor Yonglak Choi received the Ph.D. degree in Engineering from Graduate School of Soongsil University in 2001. He was Working as a Professor at Sejong Continuing graduate. He is currently a Professor of Graduate School of Software Soongsil University Since 2012. His current research interests include data modeling, software engineering and information strategy planning.