

전자상거래 웹사이트의 성능 측정 및 향상 방법

박양재
가천대학교 컴퓨터공학과

Measuring and Improving Method the Performance of E-Commerce Websites

Yang-Jae Park
Dept. of Computer Engineering Gachon University

요 약 현재 무선 인터넷 환경에서 모바일 장치를 사용하여 빠르게 웹사이트에 접근하는 것은 웹사이트 성능측정과 밀접한 관계가 있다. 웹사이트에 접속 할 때 시간이 많이 소요되면 이용자는 해당 웹사이트를 벗어나고 더 이상 접속하지 않는 특징을 가지고 있다. 이 때 웹사이트의 어떤 문제점으로 인하여 접속이 지연되는지 성능을 측정하고, 분석하여 웹사이트의 성능을 개선하여야 한다. 대부분의 전자상거래 사업 분야에서 웹기반 서비스로 개발되고 있는 상황에서 웹사이트의 성능 측정요소들 중에서 웹 페이지의 로딩 시간은 성공적인 서비스 비즈니스를 위한 매우 중요한 요소이다. 오픈소스 툴을 사용하여 전자상거래 웹페이지의 성능을 분석하여 문제점과 소프트웨어적 최적화 방법과 하드웨어적 최적화 방법을 제시하였다. 두 최적화 방법을 환경에 알맞게 적용하면 안정적이며 전자상거래 웹사이트 운영이 가능해 질 것이다.

주제어 : 웹서버, 성능측정, 네비게이션 타이밍 API, 웹성능, 웹페이지

Abstract In the current wireless Internet environment, using a mobile device to quickly access a web site is closely related to measuring the performance of a website. When accessing a website, the user has a long time to access the website and has no access to the website. In this case, the performance of the web site should be improved by measuring and analyzing the performance of the connection delay due to a problem of the web site. Among the performance measurement factors of Web sites, Web page loading time is a very important factor for a successful service business in the situation where most of e-commerce business is being developed as a web-based service. An open source tool was analyzed to analyze the performance of the e-commerce web page to present problems, software optimization methods and hardware optimization methods. Applying two optimization methods to suit the environment will enable stable and e-commerce websites.

Key Words : Web Server, Performance Measurement, Navigation Timing API, Web Performance, Webpage

1. 서론

최근에는 기업뿐만 아니라 개인들도 쇼핑몰이나 홈페이지,

개인 블로그를 운영하고 있으며, 이러한 웹 서비스 제공을 통해 수익을 창출하고자 하는 사람들이 많이 증가했다. 이러한 사람들은 더 많은 수익을 얻는데 자신의

Received 21 July 2017, Revised 26 August 2017
Accepted 20 September 2017, Published 28 September 2017
Corresponding Author: Yang-Jae Park(Gachon University)
Email: parkyj@gachon.ac.kr

© The Society of Digital Policy & Management. All rights reserved. This is an open-access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

ISSN: 1738-1916

웹사이트가 얼마나 효과가 있는지 궁금해 하고, 유입되는 이용자들의 행동패턴을 알고자 한다[1]. 그러기 위해서 인터넷 기반 사업 분야와 많은 서비스가 웹을 기반으로 개발되고 있는 상황에서 웹사이트의 성능을 측정하는 중요한 요소로 웹 페이지 로딩시간은 전자상거래 비즈니스의 성공을 위해서 매우 중요한 요소이다.

최근에는 데스크 탑 클라이언트보다는 모바일 클라이언트를 통하여 웹 페이지를 요청하는 비율이 급격히 늘어나고 있는 추세이다[2]. 1996년 발표된 HTTP 1.0이 2015년에 HTTP 2.0이 공개 되었다. HTTP 2.0은 latency를 줄여서 웹의 속도를 개선하기 위한 기술이다. HTTP Header 데이터 압축, Server push(서버에서 시작되는 전송), HTTP 1에서 존재하던 패킷을 대기행렬에 넣으면 머리에 있는 패킷은 끝에 있는 패키보다 먼저 전달되는 문제를 개선, 싱글 TCP 접속 시 병렬 페이지 로딩 구현 등의 방법을 이용하여 latency를 감소시켜 웹브라우저의 페이지 로딩 속도를 향상하고 있다[3].

웹사이트의 성능을 분석하기 위해서는 웹분석 도구(웹분석 솔루션)가 필요하다. 웹분석 방법은 웹사이트에서 직접 데이터를 수집하고 분석하는 방법은 많은 측정 지표들을 스스로 지정하고, 구현하는데 어려운 점이 많다. 웹 분석 도구는 측정할 때 사용되는 많은 환경요소와 측정요소들을 규정하고, 규칙에 의해서 웹사이트를 분석하여 리포트하도록 구현 된다.

정확한 웹분석을 하기 위해서는 웹분석 도구 잘 이해하고 활용하는 것이 필요하다. 웹분석 도구의 보고서 작성방법과 데이터를 보고서로 작성하는 방법, 웹분석 도구의 기능을 이해해야 다양한 인터넷 환경에서 측정 및 분석이 가능해 진다[4].

2. 관련연구

2.1 데이터 수집 방법

웹분석 도구는 어떻게 데이터 수집하는 방법에 의해서 분류 된다[4].

2.1.1 웹서버 로그파일 분석 방법

웹서버에 있는 접근로그(Access log)는 사용자들이 웹사이트에 접근한 모든 기록을 완전히 포함하고 있다.

대부분의 웹서버의 접근로그 파일들은 CERN과 NCSA에서 제정한 Common log format을 따르고 있다[5,6].

웹서버 로그파일을 파싱하여 데이터를 수집하는 방식으로 1993년 GetSites을 시작으로 웹로그 분석방식의 제품들이 등장하기 시작 하였으며 지금까지도 널리 사용되는 기본적인 데이터 수집방법이다. 다른 서버에 로그 데이터를 수집하고 분석서버로 전송하는 방법으로 처리도 가능하다. 하지만 페이지 태깅 방식이 등장하면서 ASP 서비스는 대부분 사라졌다.

2.1.2 패킷 스니핑 방식

이더넷은 데이터를 전송할 때 패킷이 네트워크 버스의 모든 컴퓨터에게 동일하게 전달된다. 각 컴퓨터의 LAN 카드는 유일한 MAC(Media Access Control)주소를 소유하게 되는데, 이 값으로 로컬 네트워크에서 각 호스트를 유일하게 구별 할 수 있다[7].

네트워크의 패킷(주로 HTTP 통신 패킷)을 잡아내는 패킷 스니퍼를 이용하여 데이터를 수집하는 방식이다. 1990년대 후반부터 사용된 데이터 수집방법이지만 보편적인 웹분석을 위한 데이터 소스로는 많이 사용되고 있지 않다. 스니핑 방식과 보안 탐지 솔루션은 유사한 기능을 보유하고 있기 때문에 보안 모니터링, 공격탐지 등과 연계된 분석을 할 수 있다.

많이 사용되는 로그파일 분석 방식에으로는 불가능한 다운로드 중 취소되는 비율과 취소시점, 방문자가 어떤 웹사이트 방문 한 후 다른 페이지로 이동하지 않고 곧 바로 떠나는 경우를 반송(Bounce)라고 하며, 웹 사이트 유입대비 반송의 비율인 반송율, 실시간 페이지 로딩시간이 가장 느린 페이지 추적을 할 수 있다.

전자상거래 사이트와 같이 많은 트래픽이 발생하는 웹사이트의 경우 네트워크에서 패킷을 캡처하기 위해서는 고사양의 측정 시스템 장비가 반드시 있어야만 한다. 최근에는 여러 대의 CP(Contents Provider)가 하나의 웹사이트로 구성되거나 여러 대의 웹서버가 네트워크에 분리되어 서비스하거나 콘텐츠 전송 서비스를 이용할 경우 패킷 스니핑은 불가능할 수 도 있다.

2.1.3 페이지 태깅 방식

웹페이지에 객체를 삽입하여 데이터를 수집하는 방식이다. 1997년 자바스크립트에 의한 페이지 태깅 방식이

사용되었으며 현재까지 가장 빠르게 성장한 분석 방법이다. 자바스크립트뿐만 아니라 플래쉬도 사용할 수 있다.

웹사이트는 네트워크 구성, 콘텐츠, 서버, 제공자에 관계 없이 데이터를 수집하고, 이관된 통합분석이 수행되는 장점이 있다. 이와 같은 장점으로 인해 온라인 애플리케이션 임대(ASP)와 서비스로써의 SW(SaaS)의 형식으로 많이 사용된다[8].

전자상거래 웹사이트와 같이 트래픽이 많은 경우가 적은 경우보다 정확한 측정결과를 볼 수 있으며, 타 웹사이트와 비교 분석이 가능하고, 분석 대상 웹사이트 외 경쟁 웹사이트와의 비교분석이 가능하다[9].

2.2 웹사이트 성능 분석

성능 분석의 측정지표는 측정방법, 측정기준, 측정대상, 측정 값의 속성, 제약조건을 결전함으로써 완성되며, 이후 유도측정값(Derived Metrics) 또는 계산측정값(Calculated Metrics)는 기본측정값(Basic Metrics)의 연산과 결합으로 만들어진다.

2.2.1 기본 측정지표

히트수(Hits)는 대표적인 사용자 클라이언트인 웹브라우저는 웹서버가 제공하는 정보에 접근할 때만 일어나는 클라이언트와 서버간의 통신의 최소요소이다. 페이지뷰 수(Page Views)는 히트수를 사용하면 콘텐츠의 사용량을 측정하기에 적합하지 않은 요소가 많기 때문에 페이지뷰 측정지표가 등장하게 되었다[10].

일반적으로 웹서버의 콘텐츠 이용량을 나타내기 위해서 방문수(Vists)는 사이트 방문자가 방문하여 웹사이트 이용을 끝내기까지 행위이며, 세션(Session)은 방문자가 사이트에 처음 방문하면 세션이 생성되고, 방문자가 사이트를 나갈 때 까지 유지되며, 세션타임 아웃은 30분 유지 된다.

순방문자수/방문자수(Unique Visistors/Vistors)에서 순방문자수는 일정한 기간 안에 방문자가 발생 시키는 방문(Visit)에서 중복을 제거 한 값이다. IP를 이용한 순방문자수 측정방법의 경우 중복제거를 위한 특정기간 내 방문 데이터를 원시 데이터 형식으로 보유하고 있어야 정확한 순방문자수의 측정이 가능하다는 문제점이 있으며, 쿠키기반의 순방문자수 측정인 경우에는 쿠키를 지원하지 않을 때와 쿠키 삭제 등 경우에 쿠키가 발생되

로 왜곡될 가능성이 있다[11].

2.3 웹사이트 성능분석

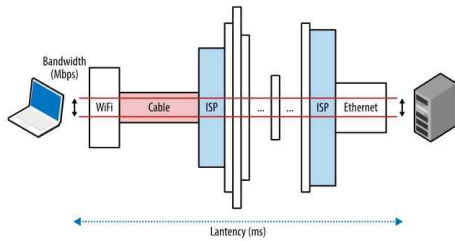
웹 성능 공학(Web Performance Engineering)은 인터넷 환경에서 모바일 장치를 이용하여 웹서버로 접근하는데 있어서 문제점인 병목현상이 발생할 수 있으며, 이 문제의 해결방법으로는 ISP(Internet Service Provider)들이 웹 사이트를 어떻게 개발해야 하는지에 대하여 다음 사항을 고려하여야 한다. 웹 개발자들은 Front-end 서버(사이트 사용자가 알 수 있는 부분인(HTML, CSS, Java Script)을 개발 하거나 Back-end 서버는 사이트 이용자는 알 수 없는 부분은 서버 단에서 실행되는 웹 프로그래밍(JSP, PHP, ASP)과 데이터베이스 프로그래밍 등을 개발하는 업무를 하는데 있어서 웹서버의 성능을 최적화 하는 방법들이 End-to-end 네트워크 환경에서 클라이언트가 웹서버에서 서비스 받는 경우 어떻게 리소스들이 어떻게 전달되는지 네트워크 의 성능을 최적화 할 수 있는 방법이 연구되고 있다.

보고서에 의하면 실험 대상 사이트는 주요 전자상거래 사이트들을 웹 사이트 분석 툴(오픈소스 기반)을 이용하여 분석하고, 장단점을 분석한 결과 상위 2000개의 전자상거래 웹사이트들이 매년 22%의 페이지로딩 시간 증가하는 추세를 나타내고 있으며, 만일 페이지를 로드하는데 1초(Second)의 시간이 더 소요된다는 것은 7%의 사용자 이탈을 의미하고, 11% 페이지 뷰가 감소하며, 16%의 고객들의 만족도가 감소한다고 한다[12].

어떤 방문자가 데이터를 요청하여 웹서버가 정확히 응답하였을 때 1 히트(Hit)로 하며 양을 정수로 계산하는 것이 히트 수이다.

2.4 웹 페이지 로딩 성능 분석

오늘 날 인터넷 서비스 제공자들은 다양한 기술을 사용하여 사용자를 네트워크에 연결할 수 있는 인터넷 액세스를 제공한다. 서비스 제공시 대역폭과 트래픽량을 계산하여 과금을 하고 있다. 그러나 인터넷 품질을 의미하는 QoS(Quality of service)의 주요 척도인 지연((Latency), 패킷손실(Packet loss), 지연변이(Delay Variation) 등은 사용자에게 알려주고 있지 않다.



[Fig. 1] Wireless network environment

[Fig. 1]과 같이 무선 네트워크 환경에서 네트워크에 접속하는 환경에서 일반적인 모바일 단말장치들은 무선 랜인 와이파이(WiFi) 또는 WCDMA 네트워크를 사용하는 3G 또는 Wibro와 LTE 네트워크를 4G 무선 네트워크 환경으로 콘텐츠 서버에 접속한다. 현재 ISP들은 최초 사업자의 서버구간을 “first mile”, 인터넷 구간을 “middle mile” 그리고 가입자 구간을 “last mile”이라고 부른다. 우리가 사용하는 네트워크 구간(가입자 구간)인 “last mile”은 다양한 네트워크 환경과 다양한 콘텐츠 서버, 여러 종류의 네트워크가 연결되어 있는 복잡한 환경이기 때문에 원거리인 경우 네트워크 지연과 많은 클라이언트들의 요청패킷이 경쟁하는 상황이 만들어진다. 이와 같이 “last-mile 문제”는 클라이언트가 네트워크를 통해서 서비스 서버에 접속하므로 대역폭과 물리적인 거리로 인한 네트워크 지연이 발생한다.

최근에 PC의 성능과 네트워크 속도의 발전으로 대부분의 웹사이트들이 빠른 응답을 제공하고 있다. Front-end 부분에 더 많은 Javascript 코드를 구현되면서 인터랙티브한 웹사이트들이 구축되고 있다. 현재 많이 사용되는 기술로는 Javascript MVC 관련 프레임 워크와 간단하지만 HTML5 기술을 이용한 웹기반 애플리케이션 사이트들도 많이 볼 수 있다. 전자상거래 사이트와 같이 많은 자원을 갖고 있는 웹사이트의 성능개선의 필요성은 더욱 증가 될 것이다. 웹 애플리케이션을 사용자가 더 빠르게 이용할 수 있다는 것은 대단한 UX가 반영된 웹사이트 보다 우수한 사용성을 제공한다.

웹 UI의 성능 개선의 목표점은 웹 페이지 로딩속도, 처리속도 등 다양한 변수가 존재한다. 웹 제작 및 디버깅 도구로는 구글에서 만든 웹브라우저인 크롬에 있는 개발자를 위한 크롬 개발자도구(Chrome DevTools)를 이용하면 HTML, Javascript, CSS의 코드생산을 효율적으로

할 수 있다.

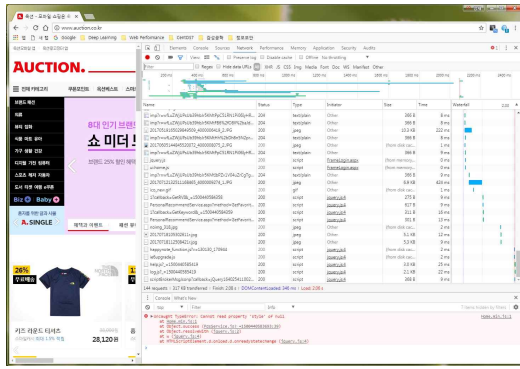
Chrome DevTools은 웹 페이지를 디자인하거나 박스 모델의 속성을 시뮬레이션 하여 어떠한 활용이 가능한지를 테스트 해볼 수 있으며 주요 기능들은 <Table 1>과 같다.

<Table 1> Sites performance measurement table

Menus	Function
Elements	Tools to analyze HTML code and modify it in real time
Resource	Tools to view the resources (images, scripts, data) used in the currently loaded web page
Network	A tool that shows the history of communications with the server, helping you find resources that do not exist
Source	Tools for debugging JavaScript embedded in web pages
Timeline	A tool to measure the performance of web pages
Profiles	A tool that measures the performance of a web page's JavaScript, CSS, etc. to determine where the bottleneck occurs. Memory status Snapshot to find memory leaks by objects
Audits	Tools to consult on ways to improve web application performance
Console	Tools to get JavaScript commands down in real time

웹 페이지 로딩 성능을 분석하기 위해서 Chrome DevTools의 Network 메뉴는 네트워크 모니터링 톨로 클라이언트의 웹브라우저와 서버사이의 흐름을 분석해주는 기능이 있다. 데이터의 송수신 내역을 쉽게 확인할 수 있으며, 웹페이지가 필요로 하는 데이터를 서버에게 받아오는 흐름을 볼 수 있다. Timeline 메뉴는 파일을 업로드하는 순서와 소요시간을 [Fig. 2]와 같이 시각적으로 나타내어 웹페이지의 성능을 측정하는 도구로서 기능이 있다.

모바일 인터넷 환경에서 모바일 장치로 데이터를 전송할 때 가입자 구간에서 발생하는 Last mile문제가 기본적으로 발생된다. 이 문제를 해결하기 위한 방법으로 클라이언트 장치와 서버간 데이터 전송시 서버 측에서 접속 풀(connection pool)을 재사용하는 방법을 적용해야 하며, 클라이언트측에서는 캐싱정책을 사용하여 필요한 객체들은 캐시를 이용하는 방법을 사용한다.



[Fig. 2] Crome DevTools Network function

2.5 웹로그 분석

오늘날 많은 기업들은 온라인 비즈니스 또는 마케팅을 위한 웹사이트를 운영하고 있으며, 자신의 기업에서 운영하는 웹사이트에서 발생한 웹로그 데이터를 마케팅에 이용할 목적으로 저장하고 분석하고 있다.

최근에는 개인도 쇼핑몰이나 개인 블로그의 운영을 하고 있으며, 웹서비스 제공을 통해서 수익을 얻는 사람들이 증가하고 있다. 웹사이트 운영자는 자신의 웹사이트가 더 많은 수익을 얻기 위해 얼마나 효과적인지 알고 싶어 한다. 해당 웹사이트에 유입되는 이용자의 행동패턴을 파악하기 위해서 웹 로그 분석 도구가 필요하다. 대표적으로 많이 사용되는 구글 애널리틱스가 있다[13].

웹사이트 운영 시 매우 중요한 부분은 웹사이트 로딩 시간(속도)이다. 웹사이트를 내부로 화려하게 잘 만들어 놓아도 클라이언트가 접속에 문제가 있다면 사이트 방문자에게 보여줄 수 없게 된다. 웹사이트에 매일 접속하는 운영자와는 다르게 처음 방문자에게 웹사이트 로딩 시간은 같지 않으므로 운영자는 주기적인 속도를 측정하여 관리해야 한다. [Fig. 3] 구글 애널리틱스 분석도구로 측정한 웹사이트 로딩 속도 측정을 나타내었다.



[Fig. 3] Measure the speed of website loading as measured by the Google Analytics analytics tool

[Fig. 3]에서 평균 페이지 로드 시간은 페이지가 모두 보여 질 때까지 소요되는 시간(초)이며, 평균 리디렉션 시간은 웹사이트에 방문자가 다른 웹 페이지가 표시되도록 하는데 소요되는 시간이다.

평균 도메인 조회시간은 IP와 도메인을 연결하는데 소요되는 시간을 의미하며, 평균 서버 연결 시간은 웹서버에 연결할 때까지 소요되는 시간, 평균 서버응답시간은 사용자가 페이지 이동을 요청할 때 서버가 응답하는 시간, 평균 페이지 다운로드 시간은 접속한 페이지가 전부 보여지기 까지 소요되는 시간을 의미한다.

3. 실험 및 결과

3.1 웹 페이지 성능 측정

웹 사이트의 성능 측정하는 방법은 종합 모니터링(사전 모니터링)방법과 에이전트 모니터링 방법이 있다.

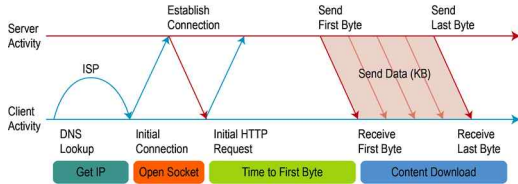
첫 번째 방법은 웹브라우저 에뮬레이션 또는 스크립팅 된 웹 트랜잭션 레코딩을 사용하여 측정되는 웹 사이트 모니터링 방법으로 액션스크립트 또는 경로는 고객 또는 최종 사용자가 사이트에서 취할 행동이나 경로를 시뮬레이션 하기 위해 만들어지며, 이와 같은 경로의 기능은 웹 사이트의 가용성 및 응답시간 측정과 같은 성능을 위해 지정된 간격으로 지속적으로 모니터링 한다. 종합 모니터링은 웹사이트를 통한 사용자의 동작 또는 탐색을 시뮬레이트 하므로 일반적으로 경로와 중요한 비즈니스 프로세스를 모니터링 하는데 가장 적합한 방법이다.

두 번째 방법은 에이전트 없는 모니터링과 에이전트 모니터링으로 분류된다. 에이전트 없는 모니터링은 서비스, 데몬 또는 프로세스가 백그라운드에서 실행되어야 하는 작업을 의미하며, 관리 서버 또는 프락시 에이전트라고 하는 에이전트가 다른 컴퓨터에서 수행된다. 이 방법은 자체작업과 관련된 소프트웨어를 설치하거나 실행이 필요하지 않으며, 원격으로 관리가 가능한 방법이다. 에이전트 모니터링은 모니터링 대상 서버에 에이전트를 설치하여 모니터링을 수행하는 방법으로 물리적, 가상 및 클라우드 환경에서 하드웨어 리소스, 운영체제, 미들웨어 및 응용 프로그램의 성능 및 가용성에 대한 요소들을 자동으로 수집하는 방법이다. 본 논문에서는 Navigation Timing API를 이용해서 종합 모니터링 방법으로 웹사이

트의 성능을 측정하였다.

TCP/IP 프로토콜에서 클라이언트가 HTTP로 웹 페이지를 요청 할 때 처리절차와 소요되는 시간을 구간별로 확인할 수 있다. [Fig. 4]는 HTTP 요청에 대한 컴포넌트들의 수행단계를 나타내었다. HTTP 요청은 DNS Lookup, Initial Connection, Initial HTTP Request, Receive First Byte, Receive Last Byte 단계로 실행된다.

[Fig. 4]와 같은 단계로 HTTP를 통해서 서버에서 클라이언트로 웹 페이지를 로딩하게 된다.

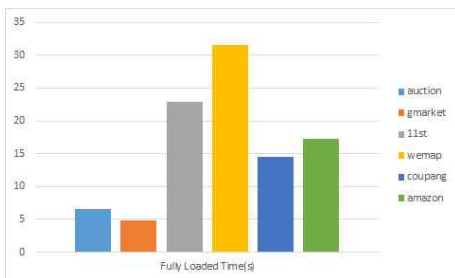


[Fig. 4] Steps for performing components for an HTTP request

HTTP 요청에 절차에 따라서 우리나라 대표적인 전자상거래 사이트인 A 사이트, B 사이트, C 사이트, D 사이트, E 사이트와 미국의 A 사이트를 W3C의 권고안인 Navigation Timing API를 이용해서 Synthetic Monitoring Tool을 개발한 대표적인 오픈 소스인 Webpagetest 를 이용하여 비교 측정하였다[14].

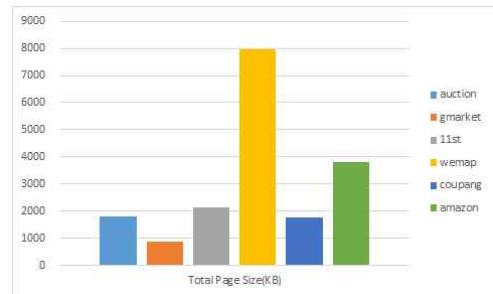
테스트 환경으로 데스크탑 컴퓨터는 CPU I7 6550, RAM 16GB, SSD 512GB이며, 네트워크 상태는 유선랜 100Mbps, 웹브라우저는 Chrome 59.0.3071.115 64비트용을 사용하여 웹성능 측정을 비교 분석 하였다.

모든 웹페이지를 로드하는데 소요되는 시간을 분석한 결과 다른 사이트에 비하여 D 사이트가 [Fig. 5]와 같이 가장 많은 시간이 소요 되었으며, B 사이트가 가장 빠른 속도가 측정 되었다.



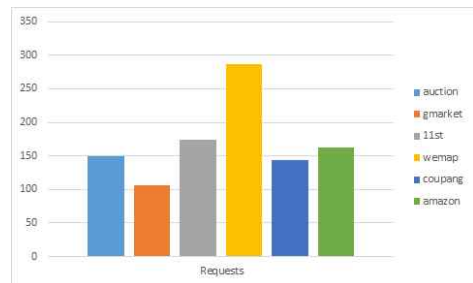
[Fig. 5] Time to load all web pages(ms)

문서 객체 모델(DOM: Document Object Model)은 객체 지향 모델로써 구조화된 문서를 표현하는 방식이다. DOM은 플랫폼/언어 중립적으로 구조화된 문서를 표현하는 W3C의 공식 표준이다. DOM 콘텐츠와 HTML 문서 등 전체 페이지 용량도 D 사이트가 가장 많아서 [Fig. 5]의 결과와 같이 웹 페이지 로드 시간이 많이 소요되었음을 알 수 있다. [Fig. 6]은 각 사이트별 웹 페이지 용량을 비교한 결과이다.



[Fig. 6] Web page size comparison for each site

각 사이트별 HTTP 요청과 리소스 요청 횟수를 사이트별로 비교한 결과는 [Fig. 7]과 같이 B 사이트가 가장 작은 횟수의 요청을 하였고, D 사이트가 가장 많은 요청 횟수가 발생되었다.



[Fig. 7] Number of HTTP requests and resource requests per site

3.2 웹 페이지 성능 향상

웹 페이지 성능 향상은 최적화 과정을 통하여 달성될 수 있다.

첫 번째 방법으로는 작성된 웹 페이지 안에 사용되는 각종 이미지들을 크기가 작은 형식으로 지정하고 압축하면 데이터 용량을 많이 줄일 수 있다.

두 번째 방법은 네트워크를 통해 리소스를 가져오는

작업 속도가 느리고 비용도 많이 소요된다. 다운로드는 서버와 클라이언트 간에 여러 번 왕복해야하므로 처리가 지연되고 페이지 렌더링이 차단될 수 있으며, 방문자의 데이터 비용도 발생한다. 모든 서버 응답은 클라이언트가 이전에 받았던 응답을 재사용 할 수 있는지 여부를 판별하는데 도움이 되는 캐싱정책을 지정하는 브라우저 캐싱을 활용하여 성능을 향상 시킬 수 있다.

세 번째 방법은 GNU gzip 유틸리티 또는 Deflate 무손실 데이터 압축 알고리즘으로 리소스를 압축하면 네트워크를 통해 전송되는 용량을 줄여서 성능을 향상 시킬 수 있다.

네 번째 방법은 스크롤 없이 볼 수 있는 콘텐츠는 다음 리소스가 로드될 때 기다려야 렌더링 할 수 있으며, 차단 리소스 로드를 지연하거나 비동기식으로 로드하거나 또는 리소스의 주요부분에 HTML에 직접 끼워 넣는다.

다섯 번째 방법은 자바 스크립트 코드를 압축하면 데이터 용량을 많이 줄여 다운로드 및 파싱 속도를 높일 수 있다.

여섯 번째 방법은 CSS 코드를 압축하면 데이터 용량을 많이 줄여 다운로드 및 파싱 속도를 높일 수 있다.

일곱 번째 방법으로는 HTML 코드(본문 자바스크립트 및 그 안에 포함된 CSS포함)을 압축하면 데이터 용량을 많이 줄여 다운로드 및 파싱 속도를 높일 수 있다.

웹 페이지는 TCP 3-Way Handshake 연결을 통하여 서버에서 클라이언트로 전달된다. [Fig. 8]과 같이 송신자와 수신자 간에 1 RTT (round-trip time)가 56 ms의 시간이 소요되었다면 첫 번째 패킷을 전송하려면 총 84 ms의 시간이 소요됨을 예측 할 수 있다[15].

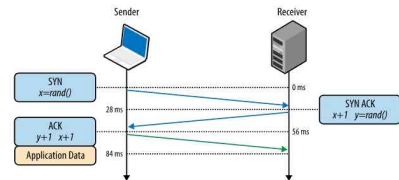
송신자와 수신자 간에 TCP 3-way handshake가 종료된 후 데이터가 전송하게 된다. TCP 대역폭 관련 파라미터들은 BDP(Bandwidth Delay Product)는 대역폭과 지연시간의 곱으로 네트워크 경로에서 전달중인 패킷의 양을 나타내며, TCP 윈도우 크기(TCP window scaling)은 TCP 연결 시 SYN 패킷 안에 수신 윈도우 사이즈(Receive window size)를 광고하는데 이 값의 범위는 64KB까지 지정하여 실제 수신 윈도우 크기는 기존 윈도우 크기와 윈도우 스케일 팩터의 곱(2^n)의 설정 할 수 있다.

TCP 소켓 버퍼 크기는 TCP 윈도우 크기를 이용하여 수신 윈도우 크기의 한계 값까지 증가 시켜도 실제 커널에 설정된 TCP 소켓 버퍼의 크기보다 크게 될 수 없다.

수신 윈도우 크기를 크게 하기 위해서는 소켓 당 버퍼의 크기를 크게 해야 한다. 네트워크는 모든 노드들의 공유 자원이며, 어떤 노드가 독점적으로 사용한다면 네트워크는 마비되고 말 것이다. 그래서 각 노드들은 혼잡 회피 알고리즘으로 전송되는 데이터 크기를 제어하고 한다.

이 알고리즘에서 참조하는 RTT(Round-trip time)가 가장 중요한 파라미터이다. 서버의 TCP 네트워크의 성능은 여러 개의 커널 파라미터를 사용하여 설정 할 수 있으나, 초기상태에서 서버는 무조건 4 세그먼트(혼잡 윈도우 크기= 4, 1 세그먼트 = 1460 바이트) 크기의 데이터를 보낼 수 있고, 그 다음에는 두 배수로 혼잡 윈도우 크기를 증가 시켜가면서 전송한다.

리눅스 커널 3.2 이상에서는 혼잡 제어 윈도우 크기가 10으로 설정되어 있다. TCP 네트워크로 데이터를 전송할 때, RTT(round-trip time) 값이 주어진다면 식(1)과 같이 전송에 필요한 시간을 계산 할 수 있다[15].



[Fig. 8] 3-Way Handshake

만일, [Fig. 8]과 같이 RTT가 56 ms일 때 데이터 크기가 64 KB를 전송하는데 필요한 시간을 계산하면, 전송 데이터 64 KB(65,536 bytes)를 세그먼트 크기(1,460 bytes)로 나누면 약 45 세그먼트가 만들어진다. 초기 혼잡 윈도우 크기는 리눅스인 경우 10으로 설정되어 있으므로, (식1)에 의하여 계산하면 168 ms의 전송 시간이 소요됨을 알 수 있고, 3-way handshake RTT(56 ms)를 더하면 264 ms의 시간이 필요함을 알 수 있다.

4. 결론

전자 상거래 웹사이트의 성능을 측정하고 분석하여 문제점과 최적화 방법을 제시하였다. 소프트웨어적 최적화 방법으로는 이미지 형식과 압축을 사용하면 데이터 용량을 크게 줄이는 방법과 웹브라우저 캐싱 기능인 모든 서버 응답은 클라이언트가 이전에 받았던 응답을 재

사용하게 하는 방법, 압축 유틸리티를 이용하여 리소스를 압축하여 네트워크 전송용량을 줄이는 방법, 스크롤 없이 볼 수 있는 콘텐츠에서 렌더링 차단하는 방법, 자바 스크립트 코드를 압축하여 다운로드 및 파싱속도를 향상시키는 방법 등을 각 사이트의 환경에 따라 측정 후 분석하여 웹 페이지를 최적화하면 최상의 서비스가 가능한 웹 페이지가 될 것이다.

하드웨어적 최적화 방법으로는 네트워크 대역폭이 충분히 큰 상황이라면 서버의 리눅스 커널을 3.2 이상으로 업그레이드를 하여 초기 혼잡제어 윈도우 크기를 10으로 설정하는 것과 Keep-alive 값을 설정함으로써 연결되었던 TCP 연결을 재사용하는 방법과 인터넷을 통하여 대용량 콘텐츠를 빠르고 안정적으로 전달해주는 네트워크 CDN(Content Delivery Network)기술을 사용하면 안정적인 전자상거래 웹사이트 운영이 가능해질 것이다.

REFERENCES

- [1] Joel J .Davis, “Google Analytics Demystified : A Hands-On Approach”, Second Edition, p.7-8, Acronpub, 2015.
- [2] Ho-Yub Lee, Jae-Min Lee, Yoo-Hyuk Lim, Eui-Seong Seo, “Analysis of Image Cache Effectiveness in Mobile Thin-Client Computing” Korea Institute of Information Scientists and Engineers pp.1736-1738, 2015.
- [3] Won-Jae Park, Young-June Choi, “ Dynamic Power Management for Webpage Loading on Mobile Devices”,Korea Institute of Information Scientists and Engineers Vol.42. No.12,pp.1623-1628, 2015.
- [4] Il-Hyun Suh, Yon-Dohn Chung “A Log-based, Interactive Web Analytics System for Backbone Networks”, Korea Institute of Information Scientists and Engineers, Vol.21, No.10, pp.651-1657, 2015.
- [5] Byoung-Yup Lee, Jun-ho Park, Jae-Soo Yoo, “ Distributed File Systems Architectures of the Large Data for Cloud Data Services”, The Korea Contents Association, Vol.12, No.2, pp.30-39, 2012.
- [6] Ji-Hong Kim, “ Agile Product Backlog Reuse : Backlog Factoring Approach ”, Journal of Digital Convergence, Vol.13, No.12, pp.171-177, 2015.
- [7] Jun-Heon Jeon, Seong-Cheol Kim, “ Energy efficient Sink node Based MAC protocol for Wireless Sensor Networks”, Journal of the Korea Institute of Information and Communication Engineering, Vol.18, No.1, pp.177-182, 2014.
- [8] Kwang-Kyu Seo, “Development Strategy of SaaS Service based on User Behavior Analysis ”, The Society of Digital Policy & Management, Vol.10, No.9, pp.73-78, 2012.
- [9] Seong-Hoon Shin, “A Study In Improving Optimal Websites For The Evolving Mobile Environment”, Communication Design Association of Korea, Vol.53, pp.109-120, 2015.
- [10] Eui-Seok Nahm, “Analysis of Transfer Characteristics and Time-delay of Control System based on Clustering Web Server”, Vol.12, No.8, pp.219-227, 2014.
- [11] Choon-Sik Park, “Metering scheme for client privacy protection ”, Journal of Digital Convergence, Vol.11, No.5, pp.291-298, 2013.
- [12] <http://venturebeat.com/2013/03/26/speed-bump-load-times-at-top-2000-e-commerce-sites-down-22-in-one-year-infog-graphic/>
- [13] Joel J.Davis, “ Google Analytics Demystified : A Hand-On Approach.” Third Edition“, 2016.
- [14] <http://www.webpagetest.org/>
- [15] Min-Ki Lim, Dong-Hee Kim, “ The Congestion Estimation based TCP Congestion Control Scheme using the Weighted Average Value of the RTT”, Digital Contents Society, Vol.16, No.3, pp.381-388, 2015.

박 양 재(Park, Yang Jae)



- 1985년 2월 : 인하대학교 전자공학과 (공학사)
- 1990년 8월 : 인하대학교 정보공학과(공학석사)
- 2003년 8월 : 인하대학교 전자계산공학과 (공학박사)
- 2001년 1월 ~ 2002년 12월 : 주식회사 이메디피아 원격의료연구소 연구소장.
- 1993년 2월 ~ 현재 : 가천대학교 IT대학 컴퓨터공학과 교수.
- 관심분야 : HCI, 영상처리, 모바일 네트워크, 감성공학
- E-Mail : parkyj@gachon.ac.kr