Ajax 기반의 웹 사이트 구축 및 성능 평가 연구 분석

황인탁*, 김진형**, 정동원*
*군산대학교 정보통계학과
**고려대학교 컴퓨터학과
e-mail : {playhit,djeong}@kunsan.ac.kr
koolmania@korea.ac.kr

Analysis of the Performance Evaluation Researches and Implementation of the Website based on Ajax

Intak Hwang*, Jinhyung Kim**, Dongwon Jeong*
*Dept of Informatics Statistics, Kunsan National University
**Dept of Computer Science and Engineering, Korea University

요약

최근 웹 정보의 증가와 웹의 효율적인 구축에 대한 요구가 많아짐에 따라 대화형 웹 페이지를 위한 웹 도구로서 Ajax를 이용하는 웹 어플리케이션들이 급격히 증가하고 있다. Ajax의 가장 큰 장점은 Refresh 기능을 이용하여 페이지 이동없이 고속으로 화면을 전환 가능하며, 서버 처리를 기다리지 않고 비동기 요청이 가능하다. 이 논문에서는 이러한 장점의 활용을 위해 Ajax의 기능을 이용한 웹 사이트 구축한 결과 보인다. 기존 Ajax 성능 평가 방법의 분석을 통해 Ajax의 성능 평가의 한계점 및 향후 연구를 위한 추가 고려사항들을 도출한다. 기존 성능 평가 연구에 대한 분석 결과는 향후 Ajax 성능 평가를 위한 기초 자료로 이용 가능하며, 또한 구축된 웹 사이트는 기존 성능 평가 분석 결과를 바탕으로 보다 다양한 성능 평가를 위한 실험 데이터로 활용될 수 있다.

키워드 : Ajax, Refresh, 비동기, 성능 평가

I. 서론

웹사이트 개발을 위해서 HTML, JavaScript, XML, CSS, ASP, JSP, PHP 등과 같은 다양한 기술들이 개발되었다[1,2,3].

그중에서 최근에 가장 주목받고 있는 Ajax는 Asynchronous JavaScript and XML의 약자로 비동기식 자바스크립트와 XML을 이용한 웹 개발 기술이라 할 수 있다. Ajax는 기존에 존재한 자바스크립트와 동적인 스크립트언어인 HTML, XML을 활용, 서버와의 비동기 통신기술과 동적클라이언트 스크립팅 기법을 포함한다. Ajax는 어떠한 종류의 ActiveX나 플러그인 프로그램을 설치하지 않기 때문에 ActiveX나 플러그인에 익숙하지 않은 사용자도 손쉽게 어플리케이션을 사용할 수 있다. 또한 Ajax가 기존의 언어와 가장 큰 차이점은 Refresh 기능을 이용함으로써 페이지 이동 없이 고속으로 화면을 전환 가능하며, 페이지 일정부분을 자동으로 업데이트 할 수 있다는 점이다. 또한 서버 처리를 기다리지 않고 비동기 요청이 가능하며, 수신하는 데이터 량을 감소시킬 수 있다.

Ajax의 장점 중 성능의 우수성을 정량적으로 보이기 위한 연구들이 진행되었다[13,14,15]. 그러나 기존의 Ajax 평가 방법들은 일부 평가 항목, 요인들만을 고려하기 때문에 다양한 어플리케이션을 개발함에 있어서 종합적인 성능평가 방법을 제안하지 못했다. 따라서 보다 다양한 요인과 평가 항목 그리고 다양한 실험 예제를 통한 실험 평가 모델 개발이 요구된다.

이 논문에서는 보다 다양한 측면을 고려한 실험 평가 개발을 위한 일차적인 연구로서, 먼저 Ajax를 이용한 실질적인 웹 사이트를 구축하여 향후 성능 평가를 위한 실험 데이터로 이용하고자 한다. 또한 기존 Ajax 성능 평가 방법의 분석을 통해 향후 보다 정확한 성능 평가를 위한 다양한 고려 사항 도출하고 보다 다양한 성능 평가 분석 모델 개발 기초 자료로

※ 이 연구에 참여한 연구자는 'BK 21 2단계 사업'의 지원을 받았음.

이용하고자 한다. 이는 Ajax의 확장을 위한 참고자료로서도 활용 가능하다.

이 논문의 구성은 다음과 같다. 제Ⅱ장에서는 관련연구로 Ajax 동작 방식과 Ajax Refresh 기법에 대하여 기술한다. 제Ⅲ장에서는 웹 사이트 구축 환경 및 기능 정의, 개발 결과에 대해 기술한다. 제Ⅵ장에서는 기존의 Ajax 성능 평가 모델 분석하고 한계점 및 향후 연구를 위한 추가 고려사항을 도출한다. 마지막으로 결론 및 향후 연구에 대하여 기술한다.

Ⅱ. 관련 연구

이 장에서는 Ajax의 동작 방식과 Ajax Refresh 기법에 대하여 기술한다.

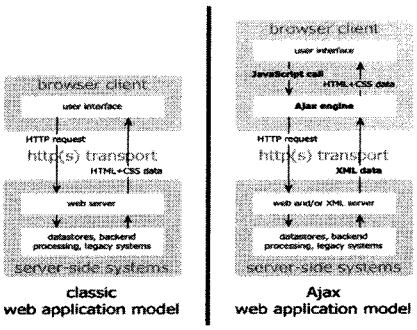
1. Ajax

Ajax는 Asynchronous JavaScript XML로서 비동기 방식으로 작동하는 다양한 영역의 기술들의 집합이다. Ajax는 XML과 CSS를 이용하여 표현하며 동적인 화면 출력 및 표시 정보와의 상호작용을 위하여 DOM(Document Object Model)[4] 및 자바스크립트를 이용한다.

Ajax는 다음과 같은 기술로 구성되어 있다[5].

- ○ XHTML과 CSS를 사용하여 표준에 근거한 프레젠테이션[6].
- ○ DOM을 통한 동적 표시 및 상호작용.
- ○ XML과 XSLT로 데이터 변환 및 제어[7].
- ○ XMLHttpRequest를 사용하여 비동기로 데이터 송수신.

그림 1은 XML과 XSLT를 이용한 데이터 교체와 조작으로 일반적인 웹 어플리케이션 모델과 Ajax 모델과 비교한 것이다.



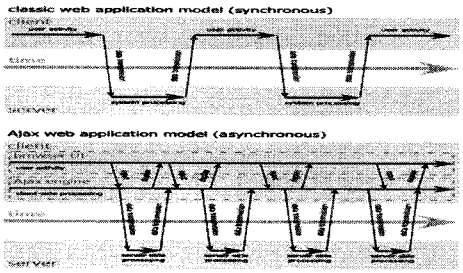
〈그림 1〉 Ajax 구조

비동기 방식인 Ajax의 방식은 사용자와 서버 사이의 중간자적인 역할을 하는 Ajax 엔진을 두어 클라이언트인 브라우저에서 Ajax엔진과의 통신에 의해서 사용자의 요청에 빠르게 응답을 한다. 즉 엔진은 XML을 항상 사용하고, 사용자의 어플리케이션과의 인터랙션을 멈추게 하지 않아 로딩 시간을 단축한다.

2. Ajax Refresh 기법

Ajax Refresh 기법은 페이지의 해당 정보를 개발자가 지정한 시간마다 자동으로 새롭게 업데이트 해주는 기능이다. Refresh 기능을 이용하면 웹을 이용하면서 Reload 없이 작업하고 이용할 수 있어 편리하게 사용 할 수 있다.

그림 2는 동기식 방식의 진행과정과 Ajax Refresh 방식인 비동기 방식의 진행과정을 보여준다.



〈그림 2〉 Ajax 동작방식

동기식 방식은 클라이언트와 서버 간에 정보를 주고 받을 때 요청에 대한 응답 시간을 기다려야 하지만 비동기식 방식은 클라이언트와 서버 간에 정보를 주고 받을 때 내부에 Ajax 엔진이 있어서 응답을 기다리는 시간 없이 다른 동작을 할 수 있다.

Ⅲ. 웹 사이트 구축

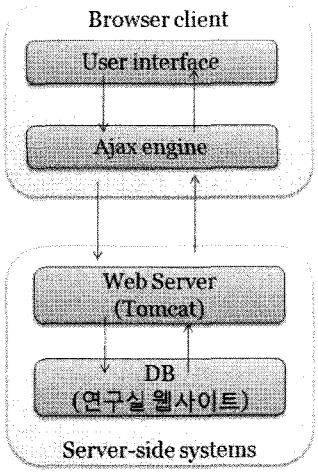
이 장에서는 Ajax 의 장점의 활용 및 적용을 위한 웹 사이트 구축에 대해 기술하며, 웹 사이트 구조, 기능 정의, 개발 환경, 개발 결과를 포함한다. 앞서 기술하였듯이, Ajax를 이용한 웹 사이트를 구축은 향후 Ajax에 대한 성능 평가를 위한 실험 환경(실험 데이터)으로 이용하기 위함이다. 이는 보다 실용적이고 현실적인 실험 환경 하에서의 Ajax와 기존 다른 웹 개발 기술과의 비교 평가를 가능하게 한다.

1. 웹 사이트 구조

클라이언트와 서버 간의 정보를 전송하는 중 기존의 웹과의 차이점은 인터페이스에서 웹 서버인 톰캣으로 정보를 전송하는 그 사이에 Ajax 엔진이 있어서 Ajax기능이 실행된다는 점이다.

그림 3은 구축한 웹 사이트에 대한 구조를 나타낸다. 구축

한 웹 사이트에서는 클라이언트와 서버 간의 정보 전송하는 중 클라이언트 측의 Ajax 엔진인 XMLhttpRequest를 통해 II.2에서 기술한 Ajax 기능을 실행한다.



〈그림 3〉 시스템 구조

2. 사용자 기능 정의

이 절에서는 구축하기 위한 웹 사이트의 주요 기능에 대하여 기술한다. 다음은 향후 실험을 위한 웹 사이트에서 지원하는 사용자 기능으로서, 단순 텍스트와 보다 다양한 자료 형태의 업로드 및 다운로드가 가능한 기능들로 구성된다.

(1) 단순 텍스트에 대한 실험을 위한 기능
- 자유게시판 : 글쓰기 기능과 수정, 삭제 그리고 Refresh 기능을 구현한다.
- 일정 : 달력형식으로 되어 있으며 일정 또한 Refresh 기능을 구현한다.

(2) 다양한 자료 형태에 대한 실험을 위한 기능
- 세미나 발표 자료 : 자료를 업로드해서, 사용자들이 다운로드 받아서 사용할 수 있게 구현한다.
- 관련 기술 자료 : 기술 관련 자료, 즉 발표자료, 기술 보고서 등 다양한 기술 관련 참고 자료를 관리하는 기능
- 세미나 동영상 자료 : 동영상 자료를 관리하기 위한 기능

3. 개발 환경

이 절에서는 웹 사이트 구축을 위해 이용된 개발 환경에 대하여 기술한다.

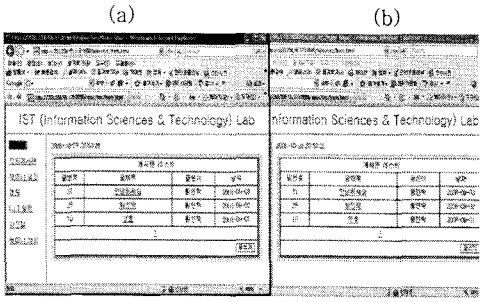
1) 클라이언트 측 환경
- 윈도우즈 XP 프로페셔널 SP3
- 인텔 코어(TM) 2 듀오 CPU E8200@2.66Ghz
- 램 2기가 바이트
- 인터넷 익스플로러 7.0

2) 서버 측 환경
- 웹 서버 : 아파치 톰캣 5.5.26
- 데이터베이스 : 오라클 10g

4. 개발 결과

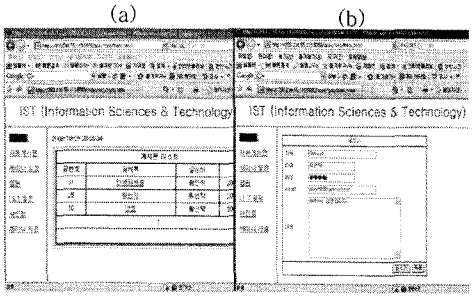
이 장에서는 웹 브라우저 2개를 실행하고 Refresh를 순차적으로 수행하여 페이지 업데이트 결과를 확인한다.

그림 4의 4(a)와 4(b)는 Refresh를 수행하기 전 자유 게시판을 나타낸 화면으로 두 개의 브라우저에서 동일한 내용을 포함한다.



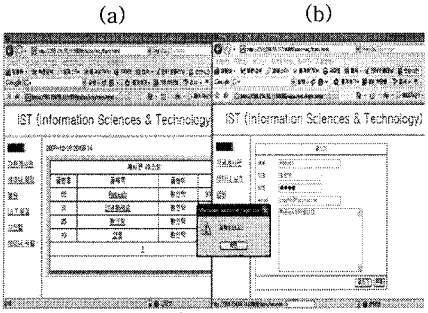
〈그림 4〉 Ajax Refresh 실험 순서1

그림 5(b)는 자유 게시판 페이지의 업데이트를 위해 새로운 게시판 글을 삽입하는 페이지를 나타낸다.



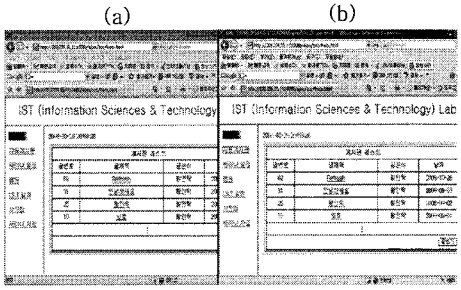
〈그림 5〉 Ajax Refresh 실험 순서2

그림 6(a)는 그림 5(b)에서 작성한 신규 글의 삽입을 반영하여 새롭게 업데이트 된 게시판 리스트를 나타낸다. 그림 6(a)의 결과처럼 자유 게시판의 업데이트 된 부분을 Ajax 엔진을 통해 별도의 Reload 과정 없이 자동으로 반영하여 나타낸다.



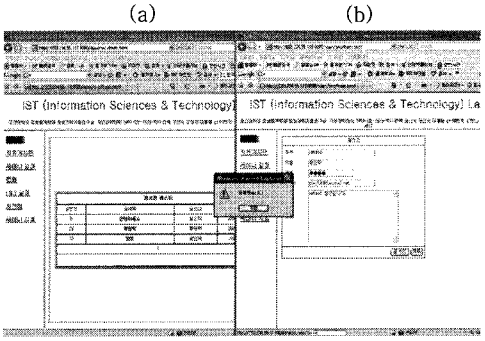
〈그림 6〉 Ajax Refresh 실험 순서3

그림 7에서 그림7(b)에서 확인 버튼을 클릭하고 확인한 결과 아래와 같이 그림 5(b)에서 입력한 정보가 그림7(b)에도 정확하게 반영되었다는 걸 확인할 수 있다.



〈그림 7〉 Ajax Refresh 실험 순서4

그림 8은 자유 게시판 페이지의 업데이트를 동기식 방식으로 했을 경우이다. 비동기 방식인 그림 6과 비교했을 경우 동기식 방식의 경우 업데이트된 내용의 반영을 위해서는 별도의 Reload 작업을 해야 한다.



〈그림 8〉 동기식 방식

Ⅳ. 성능 평가 모델 분석

이 장에서는 기존의 Ajax 성능 평가 모델 분석을 통해 기존 성능평가 모델의 장단점을 및 추가적인 고려사항을 도출한다.

1. HTML, Ajax 전송 속도 비교평가 모델

이 평가모델에서는 HTML과 Ajax의 바이트 전송 속도를 비교 평가하였다. 이 평가 모델을 이용한 비교 평가 결과로서 Ajax로 구현된 시스템은 HTML로 구현된 시스템보다 전송 시간 측면에서 68%정도 효율적이다[13].

Classification / Items	HTML (Avg)	Ajax (Avg)	Performance Increase(%)
Bytes transferred	1,737,607	460,799	73%
Time(secs)	115	78	32%
Trans. time to Web Coast	293.45	94.44	68%

〈표 4〉 HTML과 Ajax 전송 속도 비교

그러나 HTML과 Ajax 간 전송 속도 비교평가 모델의 경우 다음과 같은 한계점을 가진다. 이 평가 모델을 이용한 비교 평가 시 전송을 위한 샘플 데이터는 작은 크기로 실험을 했다. 따라서 더 정확한 실험을 하기 위해서는 일정한 크기가 아닌 작은 사이즈의 크기부터 점점 큰 크기로 실험을 해야 한다.

2. Ajax Refresh와 동기식 Page-Refresh 전송 속도 비교평가 모델

이 평가 모델의 경우, Ajax Refresh 기법과 동기식 Page-Refresh 기법을 웹 성능분석기인 Web Performance Suite 도구를 사용하여 전송 속도를 비교 한다.

전송 속도에 대한 시간은 기존의 동기식 페이지 Refresh를 사용 했을 때 보다 Ajax 비동기식 방식을 사용한 경우 61%정도가 절약된다[14].

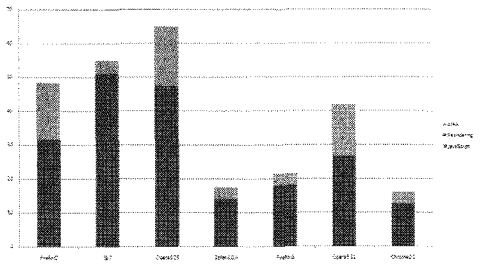
Scenario /mode	First-page Size	Typical page Size	Total bandwidth
Page-Refresh	44K	10K	210K
Ajax	47K	2.5K	81K

〈표 5〉 Ajax Refresh와 기존 Refresh 비교

하지만 이러한 평가의 경우 다양한 웹 브라우저를 이용한 실험평가가 아니므로, 다양한 웹 브라우저에 대하여 실험 결과를 보장 할 수 없다. 따라서 이러한 문제를 해결하기 위하여 다양한 웹 브라우저를 이용하여 동일한 평가가 이루어져야 한다.

3. JavaScript, Rendering, Ajax 전송 속도 비교평가 모델

이 평가 모델은 윈도우 XP 프로페셔널 SP2, 인텔 코어2 듀오 CPU 6600(2.4GHz), 데스크톱 컴퓨터 3기가 램, 아파치 웹 서버 환경에서 평가를 수행한다. 모델 평가는 JavaScript, Rendering, Ajax 조건으로 다양한 브라우저(Mozilla Firefox, Internet Explorer, Opera, Safari, Google Chrome)를 대상으로 수행한다[15].



<그림 9> Time(s) 합 결과

그림 9는 JavaScript와 Rendering 그리고 Ajax에 대한 전체적인 합에 대한 결과 그래프이다. 그래프에 대한 분포로 볼 때 Ajax에 대한 전송 시간이 JavaScript와 Rendering의 경우보다 단축된다.

세 가지 Ajax 성능 평가 모델의 분석 결과 실험에 대한 결과는 Ajax의 비동기 방식이 다른 동기식 방식보다 시간이 절약되는 것을 확인할 수 있었다. 하지만 기존의 성능 평가 모델들은 일부 평가 항목, 요인들만을 고려하였기 때문에 다양한 어플리케이션을 개발함에 있어서 종합적인 성능평가 방법을 제안하지 못했다.

정확한 Ajax의 성능 평가 및 다양한 어플리케이션 개발을 위해서는 다음과 같은 추가적인 사항들이 고려되어야 한다.

추가 고려 사항

- 다양한 크기의 데이터 전송을 통한 실험 평가
- 다양한 브라우저를 이용한 실험 평가

V. 결론

이 논문에서는 Ajax 장점의 활용 및 적용을 위해 Ajax를 이용하여 연구실 관리 웹 사이트를 구축하고, 개발 결과로서 비동기식 refresh 방법과 Ajax의 방법을 비교하였다. 또한 기존의 Ajax 성능 평가 모델들을 분석함으로써 Ajax 성능 평가의 장, 단점을 분석하고 이를 통해 좀 더 정확한 성능 평가를 위한 추가적인 고려사항을 도출하였다.

향후 연구로는 도출된 추가적인 고려사항을 반영하여 정확하고 종합적인 성능평가를 진행해야 할 것이다.

참고문헌

[1] W3C, HTML is the publishing language of the World Wide Web, 24 December 1999, http://www.w3.org/html/.

[2] W3C, Extensible Markup Language, 10 February 1998, http://www.w3.org/XML/.

[3] W3C, Cascading Style Sheets, December 1996, http://www.w3org/Style/CSS/.

[4] W3C, Document Object Model, 1 October 1998, http://www.w3.org/DOM/.

[5] Wikipedia Ajax, http://ko.wikipedia.org/wiki/Ajax/.

[6] W3C, ExtensibleHyperText Markup Language, 26 January 2000, http://www.w3.org/MarkUp/.

[7] W3C, The Extensible Stylesheet Language Family, 16 November 1999, http://www.w3.org/Style/XSL/.

[8] Samuel Juni, "The wrath of Ajax: A psychoanalytic study of Elizabethan discourse- The case of Harrington's metamorphosis," Springer, Current Psychology, Vol. 16, No. 2, pp. 99-114, 1997.

[9] Pragmatic Ajax - A Web 2.0 Primer, The Pragmatic Programmers LLC., March 2006.

[10] Yang-Won Lee, Yongcheol Suh, and Ryosuke Shibasaki, "Surveying and Geo-Spatial Information Engineering : Ajax GIS Application for GNSS Availability Simulation," Korean Society of Civil Engineers, KSCE Journal of Civil Engineering, Vol. 11, No. 6, pp. 303-310, November 2007.

[11] Jae Chun Kim, "The first American secret war : Assessing the origins and consequences of Operation AJAX In Iran, International Ares Review, Vol.9, No.1, pp.195-216, January 2006.

[12] 김성윤, 고성택, Ajax 기반 웹 스케줄러, 한국콘텐츠학회, 2006 추계종합학술대회 논문집, Vol. 4 No. 2, 2008. 11.

[13] Clinton W. Smullen III, Stephanie A. Smullen, "Modeling AJAX Application Performance," Vol. 3, No. 3, pp. 30-37, March 2008.

[14] Christopher L Merrill, "Performance Impacts of Ajax Development" 2006 Web Performance, Inc., January 15, 2006.

[15] Jordan Nielson, Carey Williamson, Martin Arlitt, "Benchmarking Modern Web Browsers", Proceedings of IEEE HotWeb 2008, pp. 1-6, Evanston, IL, USA, October 2008.