

ASP.NET Ajax를 이용한 웹 페이지 모델의 성능 분석

Performance analysis of the web site model which uses asp.net Ajax

박태흥^a 고찬^b

^a서울산업대학교 컴퓨터공학과
139-743, 서울시 노원구 공릉동 172
Tel:02-384-0285 E-Mail : heung14@hanmail.net

^b서울산업대학교 컴퓨터공학과
139-743, 서울시 노원구 공릉동 172
Tel:02-970-6705 E-Mail : chankoh@snut.ac.kr

요약

본 연구에서 웹사이트 구축시 Ajax를 사용하는 경우가 기존의 경우에 비하여 얼마만큼의 효율성을 갖는지를 비교 분석 하였다. 본 연구에서의 성능비교는, 실제로 사용되는 웹사이트의 가장 적합한 환경을 비교하기 위해 ASP를 이용한 기존의 웹사이트 모델과 ASP.NET Ajax를 이용한 웹사이트를 같은 조건에서의 샘플사이트를 구축하여 비교 분석하였다. 두 사이트 모델의 각각에 데이터를 생성, 수정, 삭제, 질의 시의 응답속도 등을 비교 분석하였다. 실험을 통한 비교 분석 결과로 기존의 웹사이트 모델보다 Ajax를 이용한 모델의 응답속도가 향상된 것으로 나타났다.

order to compare the environment which is suitable the web site which is used actually most the web site model of existing which uses ASP and the web site which uses ASP.NET Ajax constructed the sample site from same condition and comparison analyzed. Two site models a data creates, comparison analyzed amendment, elimination and question right time answer back speed etc. respectively. With the comparative analytical result which leads an experiment the web site model of existing with the fact that the answer back speed of the model which uses Ajax improves.

Abstract

The case which uses Ajax from the research which sees compared existing in case and have as how much effectiveness, comparison analyzed. Performance comparison from the research which sees, in

I. 서론

기존의 웹사이트에서 사용되는 ActiveX와 플래시 등이 동적인 느낌을 주지만 모든 브라우저에서의 호환에 제약이 있고 응답속도에 문제점이 있다. 이러한 문제점을 해결하기 위한 방법으로 Ajax가 가장 보편적인 것으로 알려져 있다. Ajax

가 일반적인언어와는 다르게 서버와 비동기식으로 통신을 할 수 있어서, 클라이언트에서 데이터가 필요할 때 언제라도 서버에 데이터를 요청할 수 있고, 이로 인하여 훨씬 더 효율적인 홈페이지 구축에 장점을 가지며 클라이언트에서 많은 것을 처리하기 때문에 서버에서 처리하는 부하도 줄일 수 있게 된다. 본 연구에서 Ajax를 사용하는 경우가 기존의 경우에 비하여 얼마만큼의 효율성을 갖는지를 비교 분석 하였다. 본 연구에서의 성능비교는, 실제로 사용되는 웹사이트의 가장 적합한 환경을 비교하기 위해 ASP를 이용한 기존의 웹사이트 모델과 ASP.NET Ajax를 이용한 웹사이트를 같은 조건에서의 샘플사이트를 구축하여 비교 분석하였다. 두 사이트 모델의 각각에 데이터를 생성, 등록, 수정, 삭제, 질의 시의 응답속도 등을 비교 분석하였다. 실험을 통한 비교 분석 결과로 기존의 웹사이트 모델보다 Ajax를 이용한 모델이 응답속도가 향상 된 것으로 나타났다.

II. Ajax

Ajax에 포함되는 요소기술은 그림1.에서 볼 수 있는 것처럼 전혀 새로운 기술이 아니라 현재 사용하고 있는 기술이다. 다만, 이러한 기술을 통합하여 사용하지 않고 부분적으로 사용했을 뿐이다. 기술의 통합을 통해 시너지 효과를 발휘하여 더욱 향상된 서비스를 사용자에게 제공함으로써 사용자 만족의 극대화를 달성하는 것이다.

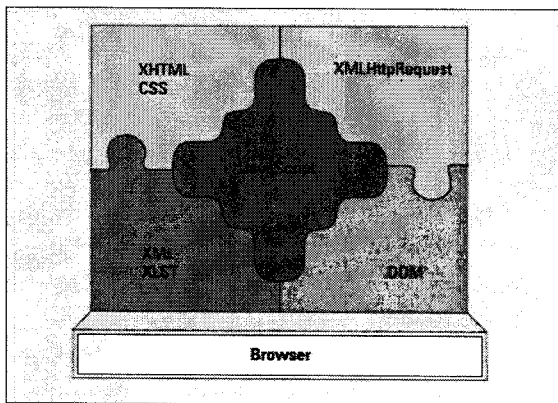


그림 1. Ajax 요소 기술

표1. Ajax 구성요소

XMLHTTP Request	웹서버와 사용자 브라우저 사이의 통신을 담당
DOM	문서의 구조를 나타내며, 브라우저 화면을 조작할 때 사용
CSS	글자 색상, 배경색, 위치와 같은 UI 용도로 사용
JavaScript	브라우저 조작을 감지하고 웹서버에 요청하여, 응답 화면 갱신

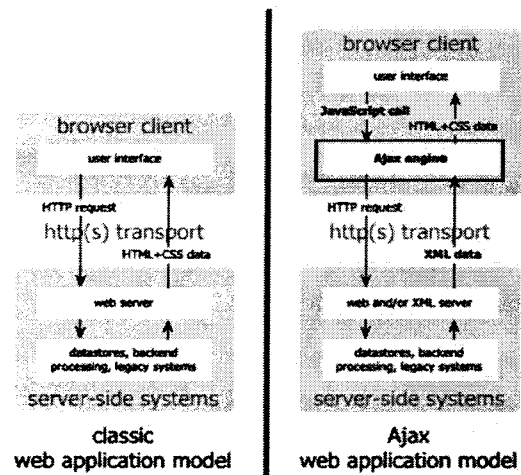


그림2. 전통적인 웹 애플리케이션 모델과 Ajax 웹 애플리케이션 모델

그림2.의 좌측 그림이 ASP/JSP 등과 같은 전통적인 웹 애플리케이션에서 처리하는 방법이고, 우측 그림이 Ajax 웹 애플리케이션 처리 방법이다. 좌측 그림에서 볼 수 있듯이 전통적인 웹 애플리케이션 모델은 사용자가 데이터를 입력하고 전송(Submit)버튼을 클릭하면 서버에서 모든 처리를 수행한다.

서버가 웹 페이지 구조를 나타내는 HTML, 웹 페이지를 표현하는 CSS, 사용자에게 제공할 데이터를 편집하여 클라이언트에 보내면 브라우저가 이를 출력한다. 즉, 서버가 HTML, CSS, 데이터를 모두 편집하고 브라우저는 단지 웹 페이지에 출력하는 역할만 한다. 그래서 사용자가 전송 버튼을 클릭한 후 서버 처리가 완료될 때까지 아무것도 할 수 없는 것이다. 우측 그림은 브라우저가 유저 인터페이스와 Ajax 엔진으로 분리되어

있으며, 서버로부터 HTML, CSS, 데이터를 받는 것이 아니라 XML 형태의 데이터만 받는다. 또한, Ajax 엔진이 서버로 받은 데이터와 HTML, CSS를 편집하여 유저 인터페이스를 제공한다. 유저 인터페이스는 서버와 연결되어 있지 않고 Ajax 엔진과 연결되어 있으며, Ajax 엔진이 서버와 연결되어 있다. 전통적인 방법에서 서버가 수행할 것을 클라이언트에 존재하는 Ajax엔진이 수행하고 있다. 이런 형태로 인해 사용자가 전송 버튼을 클릭한 후, 서버 처리가 완료할 때까지 기다리지 않아도 된다.

Ajax웹 애플리케이션은 자체적으로 통신을 해결해야 하며 이러한 통신 기능을 수행하는 객체가 XMLHttpRequest이다. 또한 통신을 하려면 이에 대한 제어가 수반되어야 한다. 이와 같이 브라우저에는 유저 인터페이스를 담당하는 브라우저 본연의 기능 이외에 다른 기능을 포함하고 있으며, 이러한 다른 기능을 통칭하여 Ajax 엔진이라고 한다.

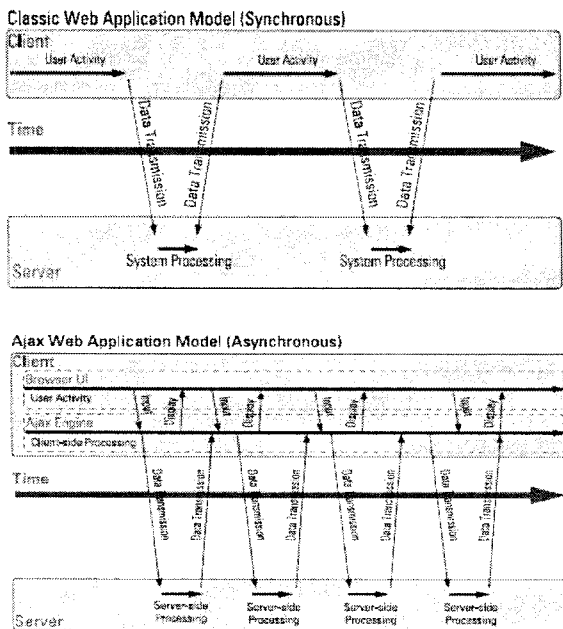


그림 3. 동기/비동기 통신 방법

그림3. 에서 상단 그림은 동기 통신 처리 흐름을 나타낸다. 클라이언트에서 사용자가 데이터를 입력한 후 전송 버튼을 클릭하면 서버로 데이터

가 전송되고 서버에서 데이터를 처리한 후 결과를 클라이언트에 보내줄 때까지 기다리고 있어야 한다. 이러한 처리가 반복되는 것을 볼 수 있다. 하단 그림은 비동기 통신 처리 흐름을 나타낸다. 클라이언트의 브라우저서 사용자가 데이터를 입력하면 Ajax엔진이 이를 받아 서버로 데이터를 전송한다. 서버가 데이터를 처리하는 동안 Ajax 엔진은 서버 처리 결과를 기다리는 한편, 사용자에게 다음 데이터를 처리할 수 있도록 제어한다. 따라서 사용자는 서버 처리에 관계없이 다음 처리를 할 수 있다.

하단 그림에서 알 수 있는 것은 유저 인터페이스와 비즈니스로직이 분리되어 있다는 것이다. 즉, 데이터를 입력하고 출력하는 것은 클라이언트에서 처리하고, 비즈니스 로직이 포함된 데이터 처리는 서버에서 처리한다. 이것은 전통적인 방법에서 웹 애플리케이션의 복잡함을 피하기 위해 의도적으로 구분하여 웹 애플리케이션을 개발했었지만, 이젠 자연스럽게 실행할 수 있게 되었다.

여기서 가장 중요한 것은 Ajax 엔진과 서버 간에 주고받는 것이 데이터뿐이라는 것이다. 전통적인 웹 애플리케이션 방법은 서버에서 HTML, CSS, 데이터 전부를 편집하여 클라이언트에 보냈지만, Ajax는 데이터만 보내고 받는다. 이것이 가능한 것은 클라이언트에 있는 Ajax엔진이 데이터만 추출하여 서버로 보내고 서버로부터 데이터를 받아 HTML, CSS를 편집하기 때문이다.

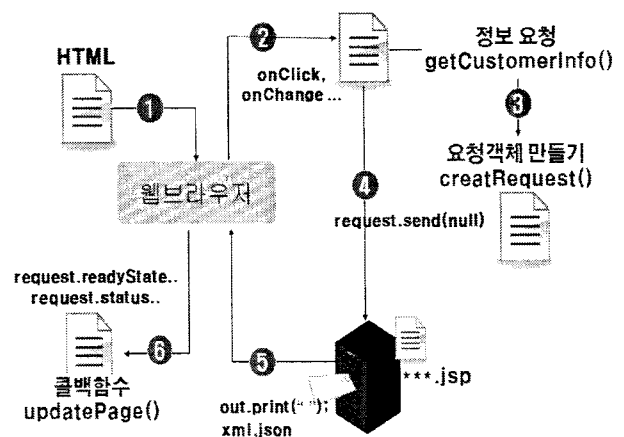


그림 4. Ajax 기본흐름

표2. Ajax의 특징

- Web 브라우저가 아닌 XMLHttpRequest 객체가 Web Server와 직접 통신함
- Web Server의 응답결과는 HTML 문서가 아닌 필요한 데이터를 담고 있는 XML이나 텍스트 문서임
- 전체 페이지를 Reload하지 않고 필요한 부분만 갱신함

표3. Ajax의 장단점

장점	<ul style="list-style-type: none"> -페이지의 이동없이 필요만으로 하는 일부만 고속으로 화면 전환됨 -서버 처리를 기다리지 않고 비동기로 처리 -ASP 및 JSP 등의 서버측 처리를 각 클라이언트 PC에 분산시켜 처리함 -유료 소프트웨어 없이도 개발 가능 -브라우저 플러그인 불필요
단점	<ul style="list-style-type: none"> -이기종 브라우저 지식(Cross Browsing Knowhow)가 필요 -자바스크립트 부분은 open 되어 있기 때문에 차별화가 어려움 -보안에 대한 주의가 필요함 -요청을 남발하면 오히려 서버 부하가 증가 -BackEnd에서 실행하기 때문에 현재 진행상황을 인식할 수 있는 메시지 필요

조회를 수행하고, TestUpdActionBean는 기존 웹을 수정하고, TestCrtActionBean는 기존 웹을 생성하고, TestDelActionBean는 기존 웹을 삭제하는 액션 빈이다. 그리고 Test2SchActionBean처럼 Test다음에 2가 붙은 것들은 Ajax를 이용한 웹 사이트 모델 방식이고, TestDTO의 자료를 임시로 저장하는 클래스이고, 마지막의 TestDAO의 데이터베이스에 접근하는 로직을 가진 클래스를 의미한다.

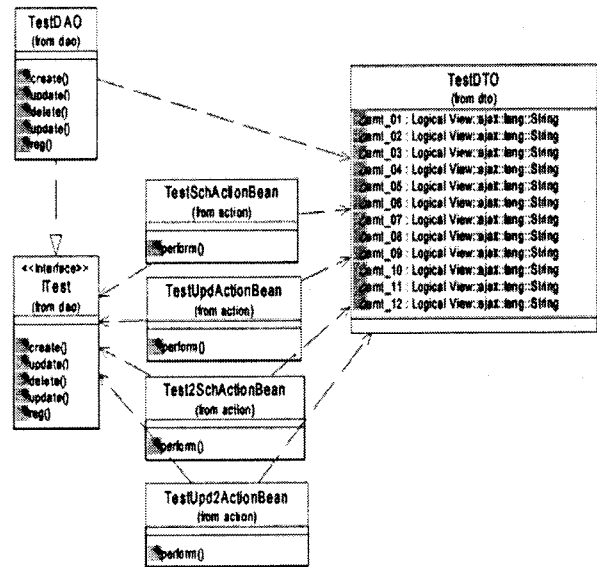


그림 5. 테스트 어플리케이션 클래스 다이어그램

그림 6.은 응답속도를 테스트하기 위한 테스트 어플리케이션 실행화면이다.

III. 측정 및 분석

동일한 자료에 대해 접근하는 기존의 웹 사이트와 ASP.NET Ajax를 이용한 웹 사이트의 응답속도를 비교함으로써 웹 사이트의 적합성과 효율성을 판단하고자 합니다. 동일한 데이터를 가지고 생성, 삭제, 수정, 조회에 대한 성능측정을 해 보고자 한다.

그림 5. 는 테스트 어플리케이션을 구성하는 클래스들의 관계를 보여주는 클래스 다이어그램으로 TestSchActionBean은 기존 웹 방식으로

추가	조회	삭제	수정	반복실행회수						
1	2	3	4	5	6	7	8	9	10	11
1	1	1	1	1	1	1	0	0	0	0
2	2	2	2	2	2	2	0	0	0	0
3	3	3	3	3	3	3	0	0	0	0
4	4	4	4	4	4	4	0	0	0	0
5	5	5	5	5	5	5	0	0	0	0
6	6	6	6	6	6	6	0	0	0	0
7	7	7	7	7	7	7	0	0	0	0
8	8	8	8	8	8	8	0	0	0	0
9	9	9	9	9	9	9	0	0	0	0
10	10	10	0	10	1	10	0	0	0	0
11	0	0	1	11	0	0	0	0	0	0
12	0	0	0	12	0	0	0	0	0	0
13	0	0	0	13	0	0	0	0	0	0
14	0	0	0	14	0	0	0	0	0	0
15	0	0	0	15	0	0	0	0	0	0
16	0	0	0	16	0	0	0	0	0	0
17	0	0	0	17	0	0	0	0	0	0
18	0	0	0	18	0	0	0	0	0	0
19	0	0	0	19	0	0	0	0	0	0
20	0	0	0	20	0	0	0	0	0	0
21	0	0	0	21	0	0	0	0	0	0
22	0	0	0	22	0	0	0	0	0	0
23	0	0	0	23	0	0	0	0	0	0
24	0	0	0	24	0	0	0	0	0	0
25	0	0	0	25	0	0	0	0	0	0
26	0	0	0	26	0	0	0	0	0	0
27	0	0	0	27	0	0	0	0	0	0
28	0	0	0	28	0	0	0	0	0	0
29	0	0	0	29	0	0	0	0	0	0

그림 6. 응답속도 측정하기 위한 실행화면

표 4. 성능분석결과(Create)

Create Record 개수	기존 웹 모델	Ajax 웹 모델
1	0.872	0.013
10	2.565	0.015
20	4.634	0.032
30	5.141	0.025

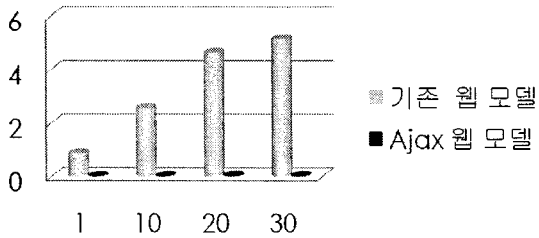
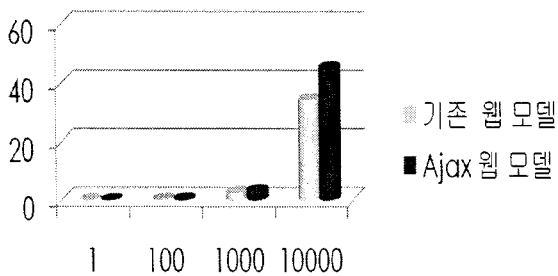


표 5. 성능분석결과(Search)

Search Record 개수	기존 웹 모델	Ajax 웹 모델
1	0.054	0.013
100	0.315	0.305
1000	2.634	3.165
10000	34.341	45.234



성능분석을 하기 위해 기존의 웹 사이트 모델과 Ajax를 이용한 웹 사이트 모델을 실행하였다. 표 4.의 내용은 데이터 생성시를 비교한 표로서 Ajax를 이용한 웹 모델이 현저히 응답속도가 빠른 것을 알 수 있었으며, 수정과 삭제시도 비슷하게 나와서 생략하였다.

그리고, 표 5.의 내용은 데이터 조회시를 비교

한 표로서 기존의 웹 모델에서 응답속도가 약간 빠른 것으로 나타났다.

이는 개발하고자 하는 웹 사이트의 성격에 따라 데이터의 생성, 수정, 삭제 시에는 Ajax를 이용하는 것이 좋고, 한번 등록을 하고 조회만을 이용하는 웹 사이트라면 기존의 웹 사이트 모델로 구성하는 것이 성능 향상시킬 수 있는 것으로 나타났다.

IV. 결론

Ajax를 사용한 웹 사이트 모델은 기존의 웹 사이트 모델보다 성능 향상을 기대할 수 있고, 사용자들의 UI를 개선하여 기존의 웹에서는 불가능했던 다양하고 편리한 UI를 가능하게 하고, 불과 결합된 Ajax는 웹 어플리케이션 개발 시 생산성 향상과 개발 후 시스템의 확장, 유지 보수를 용이하게 수행할 수 있을 것이다.

참고문헌

- [1] C. Bauer, G. King, Hibernate in Action, Manning 2004.
- [2] 김태영, Taeyo's ASP.NET AJAX V1.0 2007.12
- [3] 김종서, AJAX 기반 물류 정보 시스템 개발에 관한 연구, 중앙대학교 컴퓨터소프트웨어학과 석사학위논문 2006.12
- [4] 경영과 컴퓨터 HTTP://KYUNGC.COM.CO.KR
- [5] 강인철, 효과적인 클라이언트/서버 모델의 설계방법, 중앙정보산업대학원 석사학위논문 2001