

AJax란 무엇인가?

- Ajax란?
 - Asynchronous Javascript And XML
 - XMLHttpRequest 와 자바 스크립트를 이용한 비동기 데이터 교환
 - 새로운 기술이 아니라 현존 하는 기술의 효율적 사용
 - ActiveX나 플래시등의 플러그인 없이도 Interactive한 웹 페이지 작성
 - Reload가 필요없는 웹 페이지 작성 가능
 - 구글 맵과 더불어 2005년의 웹 이슈

AJax를 사용하는 장 · 단점 (1/2)

- 장점
 - 페이지 이동없이 고속으로 화면을 전환
 - 서버처리를 기다리지 않고 비동기 요청이 가능
 - JSP, ASP, PHP및 Perl등의 서버측 처리를 각 PC에 분산가능
 - 수신하는 데이터 양을 줄일 수 있음
 - 실시간 인터랙티브 성능이 올라감
- 플래시와 비교해서
 - 플래시가 새로운 기술을 익혀야 하는 것에 비해 Ajax는 기본 웹 기술을 그대로 사용
 - 유료 소프트웨어가 없이도 개발 가능함
 - 플러그인의 가동시간 없이 다양한 OS, Web브라우저에서 잘 동작

AJax를 사용하는 장 · 단점 (2/2)

- 단점
 - 크로스 브라우저화의 노하우가 필요
 - Ajax를 쓸 수 없는 브라우저에 대한 대책도 필요
 - JavaScript 부분은 오픈 소스이기 때문에 차별화가 어려움
 - HTTP 클라이언트의 기능이 한정되어 있음
 - 보안에 대한 주의가 불가피
 - 현재의 처리 상황에 대한 정보가 필요
 - 요청을 남발하면 오히려 서버 부하가 늘 수 있음

기존 방식의 사이트

- 웹 브라우저 자체가 요청 전송
 - 서버에 데이터 전송
- 요청을 받은 웹 서버
 - JSP/ASP/PHP등을 이용하여 요청을 처리
 - 처리결과를 HTML로 생성해서 웹 브라우저에 전송
- 응답 받은 웹 브라우저
 - 응답으로 받은 HTML을 분석하여 그 내용을 화면에 출력한다.
- 위의 과정에서 페이지 이동이 발생한다.

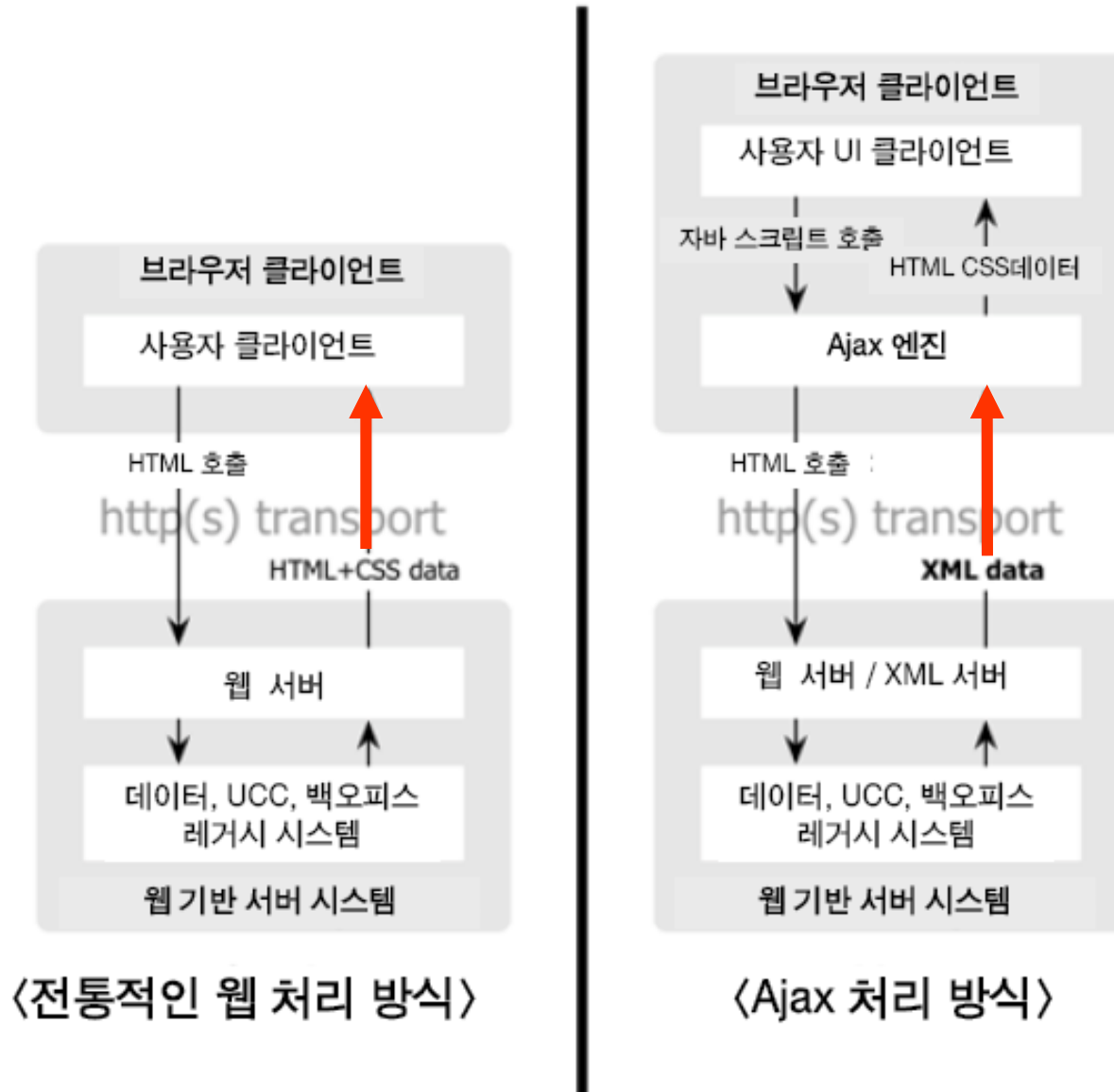
AJax 방식의 사이트

- XMLHttpRequest객체가 서버에 요청 전송
 - 사용자 이벤트 발생시 자바스크립트를 이용해서 정보를 구성한다.
- 요청을 받은 웹 서버
 - JSP/ASP/PHP등을 이용하여 요청을 처리
 - 처리결과를 XML이나 단순 텍스트로 생성해서 XMLHttpRequest에 전송
- 서버로부터 응답 도착
 - XMLHttpRequest객체가 도착 사실을 자바스크립트에 알린다.
 - 자바스크립트는 응답 데이터와 DOM을 조작해서 화면에 반영한다.
- 기존 방식과 차이점
 - 웹 브라우저가 아닌 XMLHttpRequest 객체가 웹 서버와 통신한다.
 - 웹 서버의 응답 결과가 HTML이 아닌 XML이나 단순 텍스트이다.
 - 페이지 이동 없이 결과가 화면에 반영된다.

XMLHttpRequest와 브라우저 지원 현황

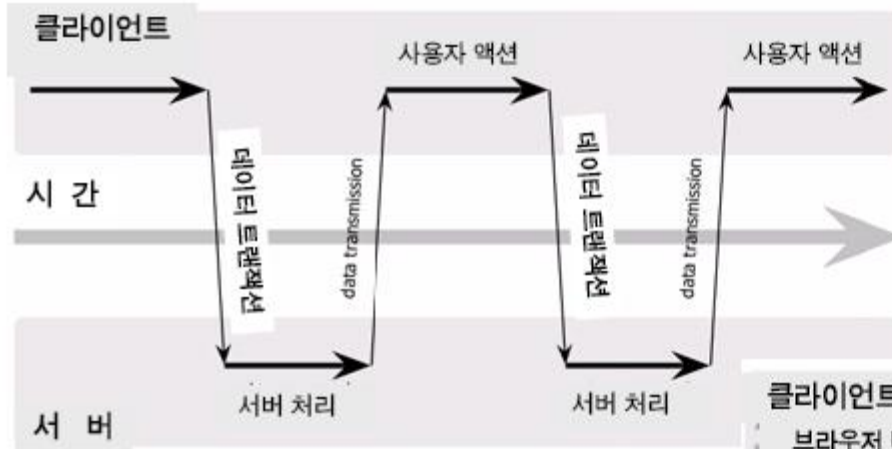
- XMLHttpRequest와 브라우저별 지원 현황
 - Microsoft Internet Explorer의 XMLHttpRequest 오브젝트는 MSXML로 최초 구현 (2000.1, IE 4.00이후)
 - Mozilla가 호환을 위해 지원 (2000. 11, Mozilla Firefox 1.0 이후)
 - 구글맵, 구글 서제스트, G-mail 서비스 개시
 - 구글의 영향으로 Machintosh 기본 브라우저인 Safari도 1.2 버전부터 지원
 - 오페라 8.0 지원시작(2005. 4)
 - konqueror 3.x 부터 지원시작

일반 Web과 Ajax의 통신방법

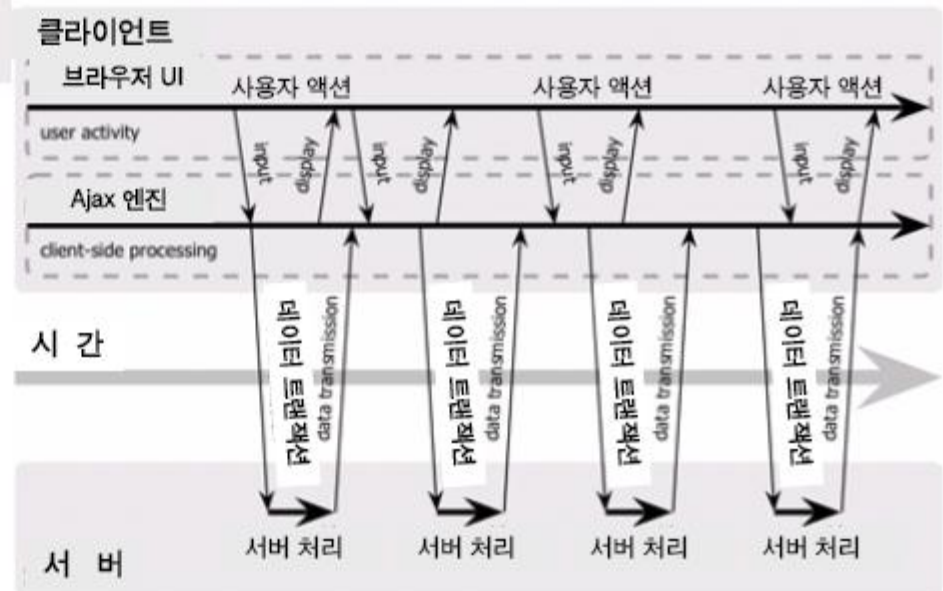


일반 Web과 Ajax의 처리방식

〈전통적인 웹 처리 방식〉



〈Ajax 처리 방식〉



Ajax의 주요 구성 요소

- XMLHttpRequest 객체
 - 웹 서버와 통신을 담당한다.
 - 사용자 요청을 웹 서버에 전송
 - 웹 서버로부터 받은 응답 결과를 웹 브라우저에 전달
- DOM
 - 문서의 구조를 나타낸다.
 - 화면 구성을 조작할 때 사용
- CSS
 - UI와 관련된 부분을 담당한다. (글자 색, 배경색, 위치, 투명도등...)
- 자바스크립트
 - 사용자 이벤트(버튼 클릭 등)가 발생하면 XMLHttpRequest 객체를 사용해서 웹 서버에 요청을 전송한다.
 - XMLHttpRequest 객체로 부터 응답이 오면 DOM, CSS 등을 사용해서 화면을 변경한다.

Ajax객체(XMLHttpRequest)의 메소드

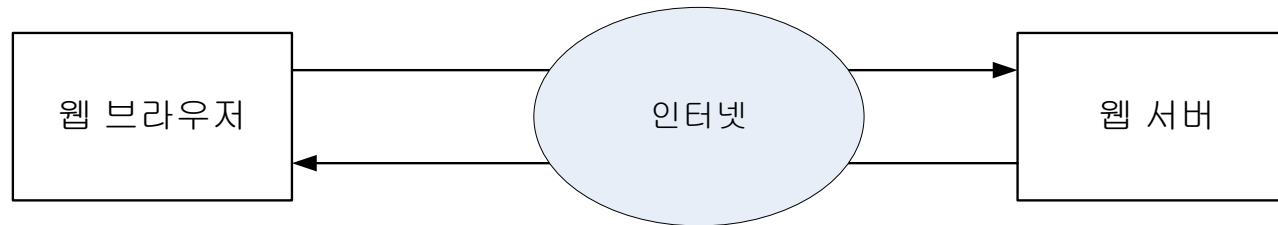
메소드	설명
<code>open(<i>method</i>, <i>url</i> [, <i>async</i>])</code>	AJAX 요청을 초기화하면서 요청 방식, 주소, 동기화 여부를 지정한다. method 인자는 http 요청 방식을 나타내며 "GET" 또는 "POST" 방식을 사용한다. url 인자는 요청할 페이지의 주소를 지정한다. aysnc 인자는 비동기 통신 여부를 나타내며 true 또는 false로 지정한다. aysnc 인자를 지정하지 않으면 true를 기본값으로 사용한다.
<code>send(<i>body</i>)</code>	AJAX 요청을 보낸다. <i>Body</i> 인자에는 요청과 함께 서버로 보낼 내용을 지정한다.
<code>abort()</code>	send() 메소드로 보낸 요청을 취소한다.
<code>getAllResponseHeaders()</code>	응답을 받은 경우 응답의 모든 헤더 정보를 문자열로 돌려준다.
<code>getResponseHeader(<i>header</i>)</code>	응답을 받은 경우 <i>header</i> 인자로 지정한 이름의 헤더 정보 값을 문자열로 돌려준다.
<code>setRequestHeader(<i>header</i>, <i>value</i>)</code>	요청을 보내기 전에 <i>header</i> 헤더 정보의 값을 <i>value</i> 로 설정한다.

Ajax객체(XMLHttpRequest)의 속성

속성	설명	읽기/쓰기
readyState	AJAX 객체의 상태를 나타내는 숫자. 처음 AJAX 객체를 생성하면 0이다. open() 메소드로 요청할 페이지 정보를 설정하면 1이 되고, send() 메소드로 요청을 보내면 2가 되고, 서버에서 응답이 오기 시작하면 3이 되고, 서버 응답이 완료되면 4가 된다.	읽기 전용
status	서버로부터 받은 응답의 상태를 나타내는 숫자. 정상적으로 응답을 받은 경우 200이고, 페이지를 찾지 못한 경우 404가 된다.	읽기 전용
statusText	서버로부터 받은 응답의 상태를 나타내는 문자열. 정상적으로 응답을 받으면 'OK'가 되고 파일을 찾지 못하면 'Not Found'가 된다.	읽기 전용
responseText	서버 응답 내용을 나타내는 문자열.	읽기 전용
responseXML	서버 응답 내용을 나타내는 XML 객체.	읽기 전용
onreadystatechange	readyState 속성이 바뀌었을 때 실행할 이벤트 핸들러를 지정한다.	읽기/쓰기

HTTP 헤더

요청-라인	GET / HTTP/1.1
메시지 헤더	Host: www.wooribank.co.kr
빈줄	
메시지 본문	



상태-라인	HTTP/1.1 200 OK
메시지 헤더	Date: Wed, 18 Oct 2006 14:03:55 GMT Server: Apache X-Powered-By: PHP/4.3.3 Connection: close Transfer-Encoding: chunked Content-Type: text/html
빈줄	
메시지 본문	<HTML> <HEAD> <TITLE>우리은행</TITLE> </HEAD> <FRAMESET ROWS="100%,*" border=0> <FRAME src=http://wooribank.com></FRAMESET> </HTML>

Ajax와 요청 헤더

- AJAX에서 요청 헤더 설정하는 법

- 요청 헤더: 웹 브라우저가 웹 서버로 보내는 부가 정보.
- [AJAX 개체].setRequestHeader([요청 헤더 이름], [요청 헤더 값])
- setRequestHeader() 메소드와 onreadystatechange 속성은 open() 메소드를 호출한 후에 사용해야 한다. open() 메소드는 AJAX 개체를 초기화하는 역할을 함.
- 부가 정보를 전달하는 용도로만 사용됨.

- 알맞은 사용자 정의 헤더 이름

- 헤더 이름은 소문자로 사용하는 것이 좋다.
- HTTP 규약에 나와있는 헤더 이름을 사용하면 안 된다.

Ajax와 응답 헤더

- AJAX에서 응답 헤더 읽어 오는 법

- 응답 헤더: 웹 서버가 웹 브라우저로 보내는 부가 정보.
- [AJAX 개체].getAllResponseHeaders()라고 하면 모든 응답 헤더의 내용을 문자열로 돌려줌.
- [AJAX 개체].getResponseHeader([응답 헤더 이름])이라고 하면 지정한 이름으로 된 응답 헤더의 값을 문자열로 돌려줌.
- getResponseHeader() 메소드로 지정한 응답 헤더가 없는 경우에 인터넷 익스플로러는 공백 문자열을, 모질라 웹 브라우저는 null을 돌려줌.

- Content-Type 응답 헤더

- 응답으로 받은 문서의 MIME 타입과 인코딩 방식을 지정.
- JSP 파일의 경우 <%@ page contentType="text/xml; charset=euc-kr" %>라고 지정.
- 응답 헤더의 값을 지정할 때는 공백 문자를 조심할 것.
- 서버 프로그램으로 XML 응답을 보내는 경우에 Content-Type 응답 헤더로 MIME 타입을 지정해야 함.

Ajax와 문자 인코딩 문제

- Ajax이 기본 문자 인코딩 방식 :UTF-8
- euc-kr로 인코딩된 응답이 깨지지 않게 하는 방법
 - Content-Type 응답 헤더로 euc-kr이라는 것을 명시적으로 표시한다.
 - Content-Type: text/xml; charset=euc-kr
- UTF-8로 인코딩 해야만 하는 경우
 - xml 파일
 - 영어, 한글 외의 다른 언어가 사용된 경우
- euc-kr로 인코딩해도 되는 경우
 - 내부적으로만 사용하고 영어, 한글 외의 다른 언어를 사용하지 않는 경우.
 - 이 경우도 사이트 관리상 어려움이 없다면 UTF-8로 인코딩하는 것이 더 좋다.
- 서버에서 문자 인코딩 변경하기

Ajax와 통신 방식

- 하나의 AJAX 개체로 여러 번 요청을 보내는 경우
 - send() 메소드로 요청을 보내기 전에 open() 메소드를 항상 실행하라.
 - open() 메소드를 실행하면 AJAX 개체가 초기화되므로 setRequestHeader() 메소드는 다시 실행해야 한다.
- AJAX로 동기 통신 방법
 - open() 메소드의 3번째 인자로 false를 지정하면 동기 방식으로 동작
 - 동기 통신 방식은 요청을 보내고 응답이 도착하기까지 아무런 작업을 할 수 없다
 - 특별한 이유가 아니라면 비동기 방식이 더 효율적이다.

Ajax와 HTTP 메소드: Get 방식

- HTTP 메소드(method)

- 웹 브라우저와 웹 서버가 데이터를 주고 받는 방식
- GET 방식: 서버에서 데이터를 가져올 때. 전송 데이터는 URI의 일부로 전송
- POST 방식: 서버로 데이터를 올릴 때. 전송 데이터는 본문(body)으로.

- AJAX에서 GET 방식으로 요청하기.

- open() 메소드의 첫 번째 인자를 'get'으로 지정
- 요청 URI를 'filename?name=value&...&name=value'와 같은 형식으로 지정하면 서버로 데이터를 전송
- URI에 사용하는 값은 encodeURIComponent() 함수를 사용하여 한글과 특수문자를 변환해야 한다.
- AJAX로 요청하는 페이지가 euc-kr로 인코딩된 경우 get 방식으로 넘겨받은 값은 UTF-8로 변환해야 한다. 서버 프로그램이 UTF-8인 경우는 변환 할 필요가 없다.

Ajax와 HTTP 메소드: Post 방식

- AJAX와 URI 설계

- URI은 한번 정해지면 사이트가 개편되더라도 바뀌지 않는 것이 좋음
- URI은 사용하기 편하고 기억하기 쉬운 것이 좋다.
- 'filename?name=value&...&name=value'과 같은 형태의 URI보다 웹 서버의 URI 재작성(ReWrite) 기능을 사용하여 단순하게 URI을 사용하는 것도 좋다.

- AJAX에서 POST 방식으로 요청하기

- open() 메소드의 첫 번째 인자를 'post'로 지정
- post 방식을 사용할 때는 setRequestHeader() 메소드로 Content-Type 요청 헤더의 값을 'application/x-www-form-urlencoded'로 지정
- send() 메소드의 인자로 전달하려는 값을 'name=value&...&name=value' 형태로 지정
- 서버로 전달하는 값은 get() 메소드와 동일하게 encodeURIComponent() 함수를 사용하여 한글과 특수문자를 변환해야 하고, AJAX로 요청하는 페이지가 euc-kr로 인코딩 된 경우 넘겨받은 값은 UTF-8로 변환해야 한다.

AJAX 응답 상태의 변화

- **status, statusText 속성**

- status 속성: HTTP 응답 상태 코드. 200이면 정상. 페이지가 없으면 404
- statusText 속성: HTTP 응답 상태 코드에 대한 설명
- status, statusText, responseXML,.responseText 속성은 응답이 도착한 후에 사용해야 함(readyState 속성이 4일 때).
- 웹 브라우저에 따라서 HTTP 규약에 나와있는 상태 코드를 지원하지 않을 수 있으므로 200, 404와 같이 널리 사용되는 상태코드만 사용할 것

- **일정한 간격으로 AJAX 페이지 요청하기**

- setInterval() 메소드로 일정한 간격으로 AJAX 페이지를 요청하는 것보다 응답 결과를 처리하는 함수에 setTimeout() 메소드를 사용하여 응답을 받은 후 일정한 시간 뒤에 다시 요청을 보내도록 하는 것이 좋음

AJAX 응답 상태의 변화

- 데이터가 변경된 경우에만 응답 내용 받기
 - 최종 수정 시간을 응답 헤더와 요청 헤더를 사용하여 AJAX와 서버 프로그램이 주고받아서 내용이 갱신된 경우에만 본문을 받으면 됨
 - 헤더의 이름은 항상 소문자로, http 규약에 정의된 이름을 피해서 정해야 한다.
- abort() 메소드
 - send로 보낸 요청을 도착하기 전에 취소

MIME 타입과 Ajax

- MIME 타입과 responseXML 속성

- responseXML 속성을 사용하려면 응답의 MIME 타입을 text/xml로 지정해야 한다.
- XML 애플리케이션인 html, rdf, smil 파일도 responseXML 속성으로 접근하려면 MIME 타입을 text/xml로 지정해야 한다. 단 rdf는 application/rdf+xml로 지정해도 된다.

- XML 기본 문법에 어긋난 XML 응답

- responseXML,.responseText 속성 둘 다 사용할 수 없다.
- 인터넷 익스플로러는 XML 파싱 중 에러가 난 경우 parseError 개체의 errorCode 속성의 값이 0이 아니다.
- 모질라 웹 브라우저는 XML 파싱 중 에러가 난 경우 응답으로 받은 내용 대신 에러를 최상위노드 이름이 parsererror인 XML 개체가 responseXML 속성에 저장된다.

MIME 타입과 Ajax

- 문서 타입 정의와 AJAX 응답

- AJAX는 XML 문서의 문서 타입 정의는 검사하지 않는다. 그리고, 문서 타입 정의에 어긋나게 작성된 XML 응답도 XML 기본문법만 지켰으면 responseXML 속성으로 접근할 수 있다.

- XML 기본 문법

- XML 문서는 <?xml ... ?> 선언으로 시작해야 한다. XML 문서는 하나의 요소로 되어있어야 한다.
- 닫는 태그를 생략하면 안 된다. 닫는 태그가 없다면 />로 시작 태그가 끝나야 한다.
- 요소의 내용이 서로 겹쳐서는 안 된다.
- 속성의 값은 따옴표를 꼭 사용해야 한다.