

2. AJAX 소개

1) 개념

◦ AJAX 개념

- Ajax (Asynchronous Javascript And XML : 비동기 방식의 자바스크립트와 XML) 는 자체가 하나의 특정한 기술을 말하는 것이 아니며 대화식 웹 애플리케이션의 제작을 위해 함께 사용하는 기술(XML, DHTML, CSS, XSL/T, JSON)의 묶음을 지칭하는 용어이다

2. AJAX 소개

2) 특징

◦ AJAX 특징

- **기존의 웹 어플리케이션 동작방식**

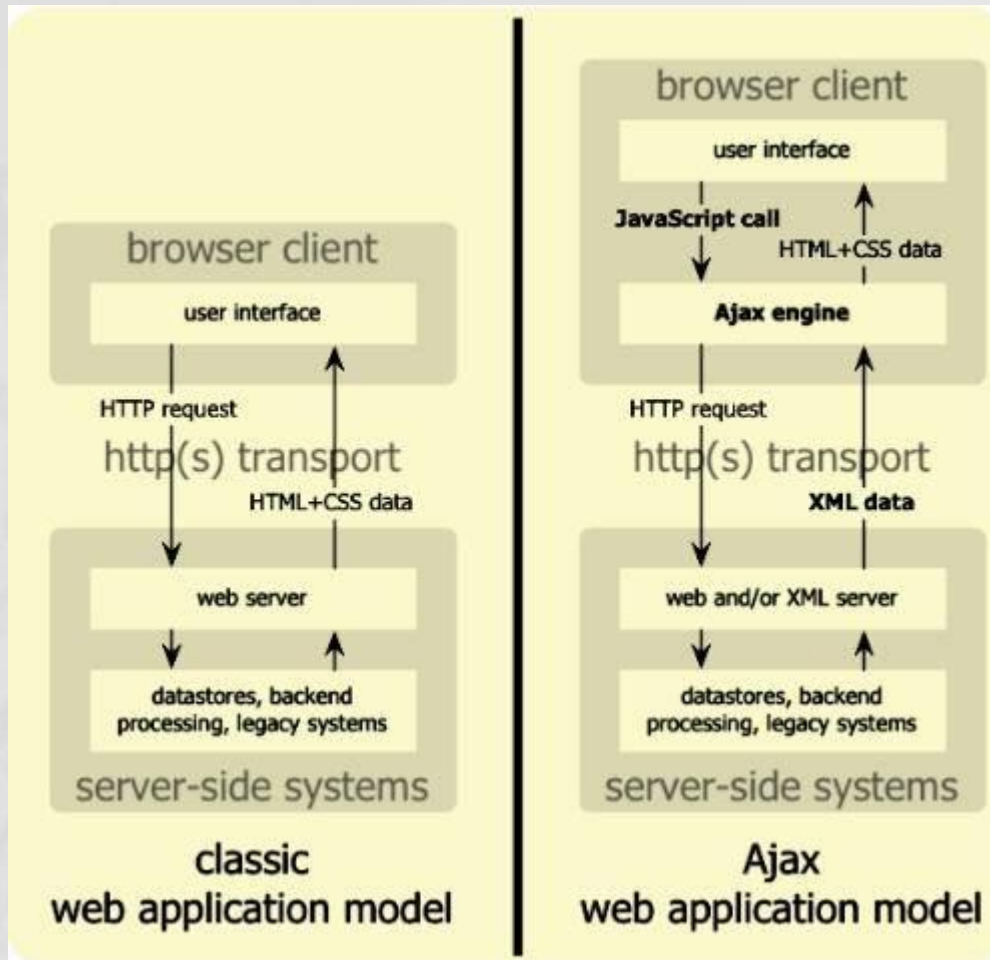
- 하나의 요청으로 웹 서버에 전달
- 웹 서버는 요청에 해당하는 응답으로 새로운 웹 페이지를 돌려줌.
- 웹 브라우저는 응답에 해당하는 웹 페이지를 다시 그려줌.
(대여푹 낭비, 사용자와 대화식의 서비스가 어려움)

- **AJAX 웹 어플리케이션의 동작방식**

- 여러 개의 비동기 요청으로 웹 서버에 전달
- 웹 서버는 요청에 해당하는 응답으로 필요한 데이터만을 돌려줌.
- 웹 브라우저의 자바스크립트를 이용하여 화면의 필요한 부분을 갱신함.
(대여푹이 줄어듦, 사용자와 대화식 서비스가 가능해짐, 웹 서버는 필요한 데이터만 생산하므로 일 양이 줄어듦.)

2. AJAX 소개

2) 특징



2. AJAX 소개

2) 특징

◦ 장점

- 페이지 이동 없이 고속으로 화면을 전환할 수 있다.
- 서버 처리를 기다리지 않고, 비동기 요청이 가능하다.
- 수신하는 데이터량을 줄일 수 있고, 클라이언트에게 처리를 위임할 수도 있다.

◦ 단점

- Ajax를 쓸 수 없는 브라우저에 대한 문제가 있다.
- Http클라이언트의 기능이 한정되어 있다.
- 페이지 이동 없는 통신으로 인한 보안상의 문제
- 지원하는 character set이 한정되어 있다.
- 스크립트로 작성되므로 Debugging이 용이하지 않다.
- 요청을 남발하면 역으로 서버 부하가 늘 수 있음.

◦ AJAX 적용사이트

- <http://map.google.com>
- <http://gmail.com>
- <http://www.wzd.com>
- <http://www.naver.com>
- <http://www.microsoft.com>

3. AJAX 사용

1) 브라우저 호환성

◦ XMLHttpRequest 객체의 사용

- Microsoft Internet Explorer4.0 이상
- Firefox1.0 이상
- Opera7.6 이상
- Safari1.2 이상
- Netscape7 이상

3. AJAX 사용

2) XMLHttpRequest

◦ XMLHttpRequest 객체 구하기

```
getXMLHttpRequest : function() {  
    // IE에서 XMLHttpRequest 객체 구하기  
    if (window.ActiveXObject) {  
        try {  
            return new ActiveXObject("Msxml2.XMLHTTP");  
        } catch (e) {  
            try {  
                return new ActiveXObject("Microsoft.XMLHTTP");  
            } catch (e1) { return null; }  
        }  
    }  
    // IE이외의 브라우저에서 XMLHttpRequest 객체 구하기  
    else if (window.XMLHttpRequest) {  
        return new XMLHttpRequest();  
    } else {  
        return null;  
    }  
}
```

3. AJAX 사용

3) 요청전송

◦ 웹 서버에 요청 전송하기

- `httpRequest.open ("POST", "/test.jsp", true);`
 - 첫 번째 인자 : HTTP 메서드를 지정
 - 두 번째 인자 : 접속할 URL.
 - 세 번째 인자 : 동기/비동기 방식을 지정.
- `httpRequest.send("id=neptune&pw=111");`
 - POST 방식일 경우 전송할 문자열을 지정.
 - GET 방식일 경우 URL뒤에 파라미터를 붙임("/test.jsp?id=neptune&pw=111")

3. AJAX 사용

4) 응답처리

- 웹 서버에서 응답이 도착하면 화면에 반영

- onreadystatechange에 콜백 함수를 지정함.
- onreadystatechange에서 명시한 콜백 함수는 readyState property값이 변경될 때 마다 호출됨

< readyState property의 상수값의 의미 >

값	의 미	설 명
0	UNINITIALIZED	객체만 생성되고 초기화되지 않은 상태 (open메서드가 호출되지 않음)
1	LOADING	open메서드가 호출되고 아직 send메서드가 불리지 않은 상태
2	LOADED	send메서드가 불렸지만 status와 헤더는 도착하지 않은 상태
3	INTERACTIVE	데이터 수신중 상태
4	COMPLETED	데이터 수신이 완료된 상태

3. AJAX 사용

4) 응답처리

< status property의 상수값의 의미 >

값	의 미	설 명
200	OK	요청성공
403	Forbidden	접근거부
404	Not Found	페이지 없음
500	Internal Server Error	서버 오류 발생

3. AJAX 사용

4) 응답처리

```
httpRequest.onreadystatechange = callbackFunction;
.....
.....

function callbackFunction() {
    if (httpRequest.readyState == 1 ||
        httpRequest.readyState == 2 ||
        httpRequest.readyState == 3) {
        // 요청에 대한 작업 중임을 표현...
    } else
    if (httpRequest.readyState == 4) {
        if (httpRequest.status == 200) {
            // 서버에서 데이터 수신완료
            // 응답 결과에 따른 작업 처리
        } else {
            alert("에러 발생 : " + httpRequest.status);
        }
    }
}
```

3. AJAX 사용

4) 응답처리

◦ 응답 데이터

- `httpRequest.responseText` : 텍스트
- `httpRequest.responseXML` : XML

◦ 동기/비동기

- 비동기 방식일 때는 `send()` 함수가 호출된 후 다음 코드가 실행되지만 동기 방식일 경우 서버와의 통신이 종료된 후 다음 코드가 실행된다.
- (주의) 동기 방식일 경우 웹 브라우저에 따라 동작 방식이 다르다.

3. AJAX 사용

5) 처리과정 시퀀스

