

다이내믹 인터페이스 설계를 위한 Ajax 기술

김은주

한림대학교 컴퓨터공학과
e-mail : ejkim628@hallym.ac.kr

Ajax Craft For Dynamic Interface Design

Eun-Ju Kim

Dept. of Computer Engineering, HallYm University

요 약

인터랙티브 웹 애플리케이션 개발 도구로 인기를 얻고 있는 Ajax 기술을 이용하여 웹에서 사용하는 도움말(Help) 기능에 적용하였다. 현재 가장 많이 사용하는 MVC 모델(Model, View, Control)은 비즈니스 로직을 서버쪽에 있는 Control 에서 모두 처리하지만 Ajax 애플리케이션은 Client 에 있는 브라우저의 JavaScript 에서 모두 처리하는 개념이다. 도움말 기능에 Ajax 기술을 사용함으로써 애플리케이션의 응답성이 향상되고 웹 서버의 부담이 줄어들게 된다.

1. 서론

불특정 다수가 사용하는 웹 사이트를 사용하는 데 있어서 도움말(Help)을 제공한다는 것은 사용자의 편리성과 편의성을 위해 필수적인 사항이다. 도움말 사용의 전통적인 방법은 웹 페이지가 바뀌어야 하기 때문에 도움말 제공에 어려움이 있었다. 이러한 어려움을 해결하기 위해 alert 창을 띄워 도움말을 제공하였다. 하지만 Ajax 는 웹 페이지가 바뀌지 않는다. 이로 인해 사용자에게 한 차원 높은 도움말과 다양한 형태의 도움말을 제공한다. 사용자가 처리하고 있는 사항을 인식하여 이에 따른 처리를 할 수 있는 것이다. 본 논문은 이러한 문제를 해결하기 위한 방법으로 웹 사이트 사용 중 에러가 발생했을 때 서버에 작성된 에러 메시지 파일을 클라이언트에 가져와 도움말을 제공하는 Ajax 기술에 대해서 논하고자 한다.

2. Ajax

Ajax 라는 말은 2005 년 2 월 18 일 Adaptive Path 라는 회사의 일원이 올렸던 블로그 글 하나가 계기가 되어 순식간에 세계로 퍼졌으며 전세계의 웹 개발자들이 Ajax 에 대한 뜨거운 관심을 보이고 있다[1].

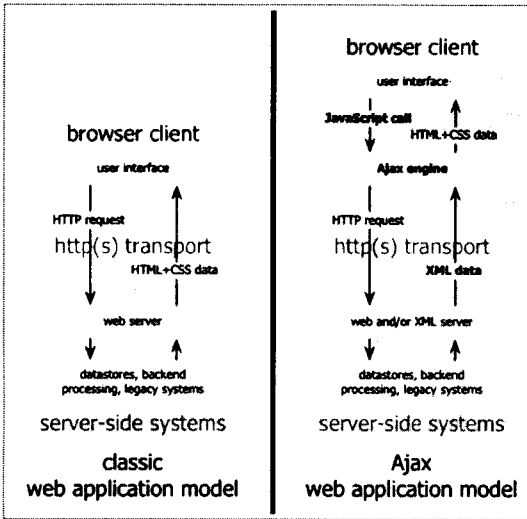
본 연구는 지방대학 혁신역량 강화사업(NURI) 문화콘텐츠(CT) 인력양성 3 차년도 사업의 연구비 지원으로 수행되었습니다.

Ajax 는 HTTP 클라이언트를 이용하는 것으로 Jesse James Garrett 가 만들었다.

현재 Ajax 기술을 사용하여 구축한 인터페이스로 유명한 것으로는 Google Suggest 와 Google Maps, Amazon A9.com(<http://a9.com>) 등이 있다.

Ajax 는 XHTML, CSS, JavaScript(렌더링 엔진을 이용), Document Object Model, XMLHttpRequest 등의 기술이 고루 섞여 대화형 웹 애플리케이션을 만들 수 있게 하는 웹 프로그래밍 기술의 복합체이며, 비동기식 자바 스크립트와 XML(Asynchronous JavaScript And XML)의 줄임말이다. 따라서 개발자들에게 잘 알려진 많은 언어가 사라진 반면 Ajax 언어는 브라우저만 있으면 Ajax 애플리케이션을 개발할 수 있으며 MS, JAVA 진영의 상황이 바뀌더라도 브라우저가 남는 한 여전히 존재할 것이다.

Ajax 는 XML 기반의 웹 서비스 언어를 사용하고 클라이언트에서는 자바 스크립트를 가지고 서버에 응답한다. 그 결과 브라우저와 웹 서버간의 데이터량이 줄어들어 애플리케이션의 응답성이 향상되고 웹 서버의 부담이 줄어들게 된다. 또한 웹에서 해당 서비스를 사용하는데 있어 별도로 프로그램을 설치(예:액티브엑스, 플래시)하거나 해당 기능을 갖춘 새 창을 띄울 필요가 없다. 일반 브라우저 화면에서 그대로 이용할 수 있으며 사용자로 하여금 직접 웹 상의 자료의 위치를 편집하는 등 커스터 마이징을 가능하게 해준다.

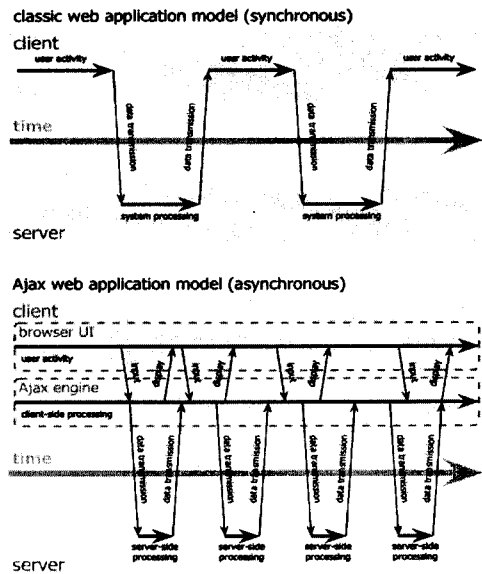


출처 : 'Ajax: A New Approach to Web Applications', Jesse James Garrett

(그림 1) Ajax 웹 어플리케이션의 모델

(그림 1)은 전통적인 웹 서비스 방식과 Ajax 방식을 이용한 웹 애플리케이션 모델이다. 기존의 방식으로 클라이언트에서 Request 를 보냈을 때 서버에서 HTML/CSS 를 보내어 응답하는 형태로 접근이 진행된다. 그러나 Explorer, Mozilla 등에서 XMLHttpRequest 를 지원하기 시작하면서 웹 브라우저에서는 XML 이라는 데이터 덩어리를 이용한 새로운 방식의 처리가 가능해졌다. Ajax 는 웹 서버에서 수신한 XML 을 JavaScript 를 통해 사용자에게 HTML 을 보여주는 기술을 보여주고 있으며 사용자와 웹 브라우저 사이에 Ajax 라는 XML 에서 HTML 로 컨버팅 계층이 하나 더 생겨남으로써 HTML 의 form 을 통한 서버와 클라이언트 교신 없이도 웹 브라우저 내부에서 자바 스크립트를 이용하여 사용자의 다양한 액션에 대응할 수 있다. 즉 서버에서 보내오는 정보와 사용자의 액션이 비동기적으로 작용하는 것이다.

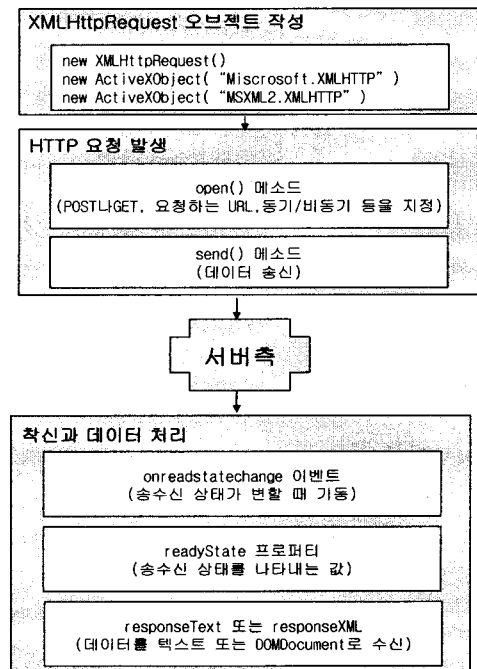
비동기라는 말은 서버 처리를 기다리지 않고 비동기 요청이 가능하다 라는 말이다. 다시 말해 이전의 동기 통신에서는 애플리케이션이 서버와의 인터랙션 (Interaction)을 필요로 할 때에 매번 브라우저가 사용자와의 인터랙션을 멈추고, 서버로부터 응답이 올 때까지 기다리고 있었다. 따라서 사용자가 그 대기 시간의 사이에 아무것도 하는 일 없이 있어야 했다. 하지만 이에 반해 비동기 통신은 사용자의 동작을 멈추는 일 없이 연속적인 작업을 이어갈 수 있다. Ajax 의 스크립트에서는 일반적으로 비동기 응답을 콜백 (callback)으로 수신한다. 즉 응답을 수신할 때 호출되는 콜백 함수를 이용하는 것으로 여기저기서 발생하는 요청과 응답을 맞추기 위한 방법인 것이다. (그림 2)는 동기방법과 비동기적 방법을 한 눈에 보여준다.



출처 : 'Ajax: A New Approach to Web Applications', Jesse James Garrett

(그림 2) 동기/비동기 방법

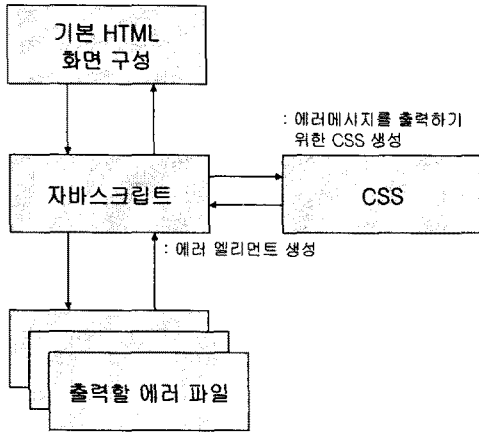
Ajax 의 핵심인 XMLHttpRequest 에 의한 송수신 흐름은 (그림 3)을 통해 해결된다. XMLHttpRequest 오브젝트를 생성하여 HTTP 요청을 준비한 후 서버에 접속하여 데이터 및 요청을 보내고, responseText 또는 responseXML 프로퍼티로 수신하면 된다[2].



(그림 3) XMLHttpRequest 에 의한 송수신

3. Ajax 를 이용한 도움말 기능

웹 페이지에서 데이터를 입력하고 서버 및 버튼을 클릭하면 필수 항목 입력 여부 체크와 같이 기본적인 에러 체크를 하고 alert 창을 통해 메시지를 출력한다. 이와 같이 에러 메시지를 자바스크립트에 포함시키는 것은 좋은 방법이 아니다. 즉 시스템 전체에 공통으로 사용하는 에러 메시지를 사용해야 일관성 있는 메시지를 제공할 수 있다.



(그림 4) Ajax 방법을 적용한 도움말 모델

(그림 4)은 Ajax 방법을 적용하여 도움말 기능을 설계한 모델이다. 우선 HTML 파일에는 일반 HTML 파일과 차이가 없게 문서를 작성하면 된다. 이때 에러 메시지에 대한 엘리먼트는 작성하지 않는다. 에러 메시지에 대한 엘리먼트는 자바스크립트에서 작업해 주면 되고 에러 메시지에 대한 출력 형식을 CSS 파일에 작성해 두면 된다.

```

<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="ko" xml:lang="ko">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link rel="stylesheet" href="help.css" type="text/css" />
<script language="javascript" type="text/javascript" src="callBackXMLHttp.js">
</script>
<script language="javascript" type="text/javascript" src="help.js">
</script>
<script language="javascript" type="text/javascript" src="commFunction.js">
</script>
<title>... </title>
</head>
    
```

(그림 5) HTML 파일 선두 부분에 삽입할 내용

위에서 보면 CSS 파일과 자바스크립트 파일을 호출

하기 위하여 스크립트 태그를 적용한 것을 볼 수 있다. 이때 callBackXMLHttp 스크립트와 commFunction 스크립트 파일도 같이 호출한다. (그림 6)는 초기 화면의 HTML 파일이다.

```

<div id="dataarea">
  <div class="lineitem">
    <span class="itemdesc">이름</span>
    <span class="dataentry">
      <input type="text" id="ename" size="10" >
    </span>
  </div>
  <div class="lineitem">
    <span class="itemdesc">취미</span>
    <span class="dataentry">
      <input type="text" id="ehobby" size="20" >
    </span>
  </div>
</div> <!-- id="dataarea"-->
    
```

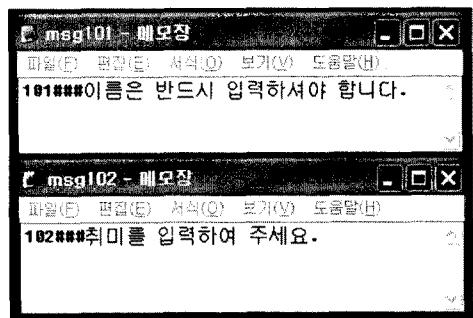
(그림 6) 기본 화면 구성 HTML 파일

```

/* 에러 메시지 CSS */
#msg101 {
  position: ...
  width: ...
  background-color: ...
  border: ...
}
#msg102 {
  position: ...
  width: ...
  background-color: ...
  border: ...
}
    
```

(그림 7) 에러 메시지에 적용할 형식 지정

(그림 7)은 에러 메시지를 출력하기 위한 CSS 이다. 자바스크립트에서 msg101 과 msg102 의 id 엘리먼트를 생성하게 되며 생성한 엘리먼트를 CSS 가 표현하게 된다. 아래 그림은 메시지를 입력한 텍스트 파일이다. 에러 메시지를 알려주기 위해서 텍스트로 메시지를 작성하면 된다.



(그림 8) 텍스트로 작성된 에러 메시지

```

-----
// showMessage: 오류 메시지 파일 수신
//-----
function showMessage(xmlHttp) {

    var xmlHttpData = xmlHttp.responseText; //이름, 취미 메시지
    var msgContent = xmlHttpData.split("###"); //###으로 Split

    //에러 코드 결합
    var messageCode = "msg" + msgContent[0]; //에러 코드 결합

    commRemoveChild(messageCode); //에러 메시지 삭제

    //에러 메시지 <div id="msg101,102"><div> 생성
    var messageDiv = document.createElement("div");
    messageDiv.id = messageCode;

    //<div id="msg101,102">에러 메시지</div> 결합
    var messageData = document.createTextNode(msgContent[1]);
    //에러 메시지
    messageDiv.appendChild(messageData);
    //서제스트 데이터 출력
    document.body.appendChild(messageDiv);
}
    
```

(그림 9)자바스크립트로 작성된 에러메시지 파일수신

위 프로그램은 서버에서 파일이 수신되면 실행되는 callback 함수이다. 서버에서 에러 메시지를 수신하였으므로 이를 웹 페이지에 출력해야 하므로 에러 메시지를 처리하기 위한 일련의 과정을 포함하고 있다. 서버에서 수신한 파일을 '###'을 기준으로 에러 코드와 에러 메시지를 분리한다. 첫번째가 에러 코드이고 두번째가 에러메시지이다. 에러 메시지가 출력하기 위한 HTML 엘리먼트가 <div id="msg + 에러번호">로 구성되어 있으므로 에러 코드를 결합한다. 만약 에러 메시지가 출력되어 있으면 이를 삭제하기 위해 commRemoveChild()함수를 사용한다.

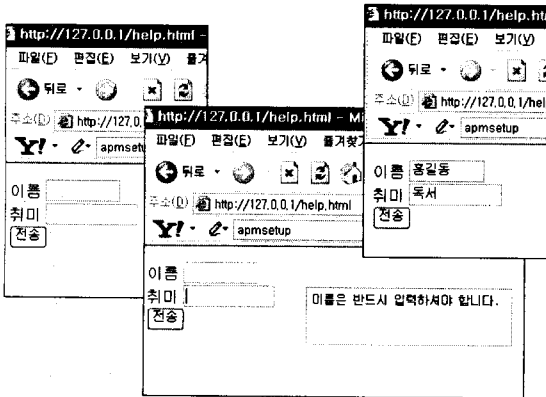
데이터를 입력하지 않고 커서가 다음 데이터 입력탭으로 이동했을 때 에러 메시지가 출력되는 모습을 보여준다. 정상적으로 데이터를 입력하였을 경우는 에러 메시지 표시 없이 실행이 된다. 각 항목에 데이터가 입력되지 않을 때에는 openSendStatus()함수를 이용하여 출력하고자 하는 에러 메시지를 열어주면 된다.

4. 결과 및 향후 연구

Ajax 의 기술에 대해서 모두 다루지는 않았지만 Ajax 는 브라우저 측으로부터 프로그래밍 가능한 HTTP 통신 기능이라는 면에서 분명히 광범위하게 쓰여 나갈 것이며 DHTML 의 연장선상에 있는 진화형으로서 더욱 발전하리라 본다. 본 논문은 이러한 차세대 웹 개발 도구인 Ajax 를 이용하여 웹 사이트에서 주로 사용되는 도움말 기능을 구현하였다. 서버로부터 데이터를 매번 불러 들여와 처리하는 시간을 단축하였고 이러한 시간의 단축은 실시간 인터랙티브 성능이 올라감을 실감할 수 있다. 또한 새로운 기술을 습득해야 하는 어려움을 없애고 기존의 기술을 사용하여 구현할 수 있다는 장점이 있었다. 향후 Ajax 기술을 활용하여 데이터 방송에서 사용할 새로운 콘텐츠 개발에 응용해 보고자 한다.

참고문헌

- [1] Ajax 입문, 한빛미디어, 타카하시토시로 지, 이창신 역, 2006 년
- [2] Ajax 활용, 가메출판사, 김영보 저, 2006 년
- [3] <http://kr.sun.com/developers/ajax/>



(그림 10)구현 화면

구현화면에서 보면 초기의 화면과 첫번째 항목에