CIS 4004: Web Based Information Technology
Spring 2013

Introduction To AJAX – Part 2

Instructor : Dr. Mark Llewellyn
markl@cs.ucf.edu
HEC 236, 407-823-2790

Department of Electrical Engineering and Computer Science
University of Central Florida
An RSS Headline Reader

• In the previous set of Ajax notes, we developed a wrapper class to deal with the XHR object to handle the request/response to the server.

• We then looked at an example that returned the contents of a text file on the server to the client’s browser window using the asynchronous nature of Ajax.

• In this example, we’ll look at the data coming from the server, not as text, but as XML data.

• Unlike, the text data, which does not require parsing to deal with on the browser-side, we’ll need to parse the XML data in order to do something with it on the browser-side.
An RSS Headline Reader

• XML data is made available on the Internet in many forms. One of the most popular is the RSS feed, a particular type of XML source usually containing news or other topical and regularly updated items.

• RSS feeds are available from many sources on the Web, including most broadcast companies and newspaper publishers, as well as specialist sites for all manner of subjects.

• We’ll develop an Ajax application to take a URL for an RSS feed, collect the XML, and list the titles of the news items contained in the feed.

• A typical RSS feed might look similar to the one shown on the next page.
Fertilizer plant explosion kills 5 to 15 people

The blast in the Texas town of West injured more than 160. "Criminal activity" is not apparent but hasn't been ruled out.

FULL STORY  Email this  Add to del.icio.us

Digg This!  Share on Facebook  Stumble It!

Mailbox may have saved his life

Amputees never say, 'I can't'

Jothy Rosenberg, who lost a leg when young, says those who lost legs in the Boston blast face big challenges—of both body and especially mind—but will succeed better than they imagine.

FULL STORY  Email this  Add to del.icio.us  Digg This!

Share on Facebook  Stumble It!

Bloody EMT chief: 'Roof came in'
An RSS Headline Reader

• Notice that the RSS feed shown on page 5 is using RSS version 2.0.

• The versions of RSS differ quite a bit, but for the purposes of our example RSS reader, we only care about the `<title>` elements for the individual news items. This element remains essentially unchanged from RSS version to version.

• Our page will need to contain an input field (read form) for the user to enter the URL of the RSS feed and a button to instruct the application to collect the data.

• We’ll also use a `<div>` element in which to display the parsed data.

• This part of the markup is shown on the next page.
An RSS Headline Reader

<h3>An Ajax RSS Reader</h3>

<form name="form1">
  URL of RSS feed: <input type="text" name="feed" size="50" value="http://">
  <input type="button" onClick="getRSS()" value="Get Feed"><br><br>
  <div id="news" class="displaybox"><h4>Feed Titles</h4></div>
</form>
An RSS Headline Reader

An Ajax RSS Reader

URL of RSS feed: [Text Input]

Get Feed

Feed Titles
An RSS Headline Reader

• Much of the actual RSS reader JavaScript is very similar to that which we developed in the previous set of notes, so I won’t put all of that code in the notes. Instead, I’ll focus on the parts which are different and involve the XML and server-side helper function.

• This time, since we are not using the responseText property of the XHR object, we’ll be using the responseXML property.

• We’ll use the data in the responseXML property to modify the DOM of our HTML page to show the new item’s titles in a list within the page’s `<div>` element. We’ll put each title into it’s own paragraph element, which we’ll construct dynamically and append to the DOM.
An RSS Headline Reader

• The callback function in this case; will perform the normal duties of checking the XHR object `readyState` and `status` properties, but we’ll also task it with the following responsibilities:
  – Clearing out any elements displayed in the `<div>` from a previous request.
  – Parse the incoming XML to extract the title elements.
  – Construct the DOM elements to hold and display the results.
  – Apply the CSS to these new elements to change how they are displayed in the browser.
An RSS Headline Reader

• To remove the DOM elements installed by a previous request, if they exist, we’ll identify the `<div>` element by using its id and then use the `hasChildNodes()` DOM method, looping through and deleting the first child node from the `<div>` element each time until none remain.

• Next, we’ll process the `<title>` XML elements. To parse the XML content to extract the item titles from the RSS feed, we’ll build an array `titleNodes` from the XML data stored in the XHR object’s `responseXML` property.

• The code snippets for both of these tasks are illustrated on the next two pages.
```javascript
//callback function
function cbgetRSS() {
    if (http.readyState === 4) {
        if (http.status === 200) {
            // first remove the childnodes presently in the DM
            while (document.getElementById('news').hasChildNodes()) {
                document.getElementById('news').removeChild(document.getElementById('news').firstChild);
            }
            var titleNodes = http.responseXML.getElementsByTagName('title');
        }
    }
}
```
An RSS Headline Reader

```javascript
for(var i = 1; i < titleNodes.length; i++)
{
    var newtext = document.createTextNode(titleNodes[i].childNodes[0].nodeValue);
    var newpara = document.createElement('p');
    newpara.className = "title";
    var para = document.getElementById('news').appendChild(newpara);
    newpara.appendChild(newtext);
}
```
An RSS Headline Reader

• Because of security constraints built into the XHR object, you cannot call and RSS feed directly. Rather you must use a script have a URL on our own server, and have this script collect the remote XML file and deliver it to the Ajax application.

• In this case, we don’t require the server-side script to modify the XML file, but simply route it back to us via the responseXML property of the XHR object.

• The server-side script is acting as a proxy because it is retrieving the remote resource on behalf of the Ajax application.

• The next page illustrates this PHP script.
An RSS Headline Reader

```php
<?php
$mySession = curl_init($_GET['feed']);
curl_setopt($mySession, CURLOPT_HEADER, false);
curl_setopt($mySession, CURLOPT_RETURNTRANSFER, true);
$out = curl_exec($mySession);
header("Content-Type: text/xml");
echo $out;
curl_close($mySession);
?>
```
An RSS Headline Reader

- The PHP script uses the cURL PHP library. This is a set of routines for making Internet file transfer easier to program. Note that this library is not automatically installed in most PHP installations, so you will need to activate this library manually.

- A description of cURL is way beyond what we want to get into here, suffice to say that this short script first retrieves the URL of the requested RSS feed by referring to the `feed` variable sent by the Ajax application.

- The two lines that call the `curl_setopt()` function declare, respectively, that we don’t want the headers sent with the remote file, but we do want the file contents.

- The `curl_exec()` function then handles the data transfer.

- Once that’s done, we simply add an appropriate header using the PHP `header()` command and return the data to the Ajax application.
An RSS Headline Reader

URL of RSS feed: http://rss.cnn.com/rss/cnn_latest.rss

CNN.com Recently Published/Updated

North Korea outlines exacting terms for talks with the U.S.

Volunteer firefighters step up despite personal tragedy, danger

Healing in Boston as investigators seek 2 seen near bombing

Opinion: Amputees never say 'I can't'

Opinion: Should CIA be targeting terrorists?

German workers worried after delivery of poison-spiked bread rolls

Explosion hits fertilizer plant north of Waco, Texas

Pakistani court rejects Musharraf bail, raising possibility of arrest

US now Japan's number one export country

Don't let allergies stop you from traveling

Founder of French breast implant company goes on trial
An RSS Headline Reader

URL of RSS feed: https://www.competitivecyclist.com/za/CCY?PAGE=NEW_PRODUCTS_RSS

Get Feed

New Products

2013 Gore RideOn Professional Derailleur Cable Kit

2011 SRAM Red Rear Derailleur

2013 Fi'zi:k Arione Versus K:IUM Saddle

2009 Fi'zi:k Vitesse Tri Carbon Saddle

2012 Shimano Dura Ace ST-7900 STI Shifters

2012 Continental Ultra Sport Hometrainer Tire - Clincher

2011 WTB ExiWolf Race Tire - 26in

2013 Q-Tubes 27.5in Presta Mountain Tube

2012 Easton Haven 26in Wheelset

2012 SciCon Phantom 230 Roller 2.0 Saddlebag

2012 Louis Garneau Middle Race Bag