

## Programming in AJAX

We consider:

Data formats: XML vs. JSON

Raw code vs. libraries (for example Scriptaculous)

### *Retrieving Data using XML*

```
var xmlDoc;
function getXMLdoc(){
    var http;
    try{
        http = new XMLHttpRequest();
    } catch (e) {};
    if (!http){
        http = new ActiveXObject("Microsoft.XMLHTTP");
    }

    http.open("GET", "responseXML.php?" + Math.random(), true);
    http.onreadystatechange = function(){
        if (http.readyState==4){
            if (http.status==200){
                try {
                    xmlDoc = http.responseXML;
                } catch (err){
                    alert('XML Error: '+err+'\n'+http.responseText);
                    return;
                }
                alert("Loaded xmlDoc:" + xmlDoc);
            }
            else
                alert(http.responseText)
        }
    }
    http.send(null);
}
```

Note that we use the responseXML method of the XMLHttpRequest object. This causes the response to be parsed as an XML document.

The good news is:

- we can use the DOM methods
- we can include a DTD and ensure validation of the XML

The bad news is:

- We must get the mime-types right
- We must use a DTD if we want to use the getElementById method.
- The method getElementById doesn't work in IE.

## ***Retrieving Data Using JSON***

You can use `responseText` instead of `responseXML`. This means that the text will not be parsed as XML and it can take almost any format. A popular and useful format is JSON. This is a JavaScript format that can be evaluated directly in JavaScript.

```
...
try {
    resp = eval("(" + http.responseText + ")");
} catch (err) {
    alert('JSON Error: ' + err + '\n' + http.responseText);
    return;
}
...
```

JSON formats arrays with square brackets and objects with curly brackets. Here is an example of JSON expressions:

```
var x=
{firstName:'Harry',lastName:'Potter',
  house:'Gryphendor',
  subjects:['Herbology','Potions','History of Magic']};
alert(x.firstName);
alert(x.subjects[1]);
```

We can transmit a string representing an object or an array as the output from the server script. The `eval` statement will interpret the object directly as a JavaScript object.

Converting into a JSON string is relatively trivial. For example it can be done in a few lines of PHP or Perl. There are JSON libraries for all of the popular languages.

### ***On JSON versus XML.***

JSON takes up less space.  
JSON is simpler.  
JSON is easier to parse.  
JSON is good enough

on the other hand

XML has validation.  
XML has tools such as XSLT  
XML has more features

# Scriptaculous

The Scriptaculous library includes functions for doing "flashy" AJAX effects and transitions.

- Using a library means not having to worry about browser incompatibility.
- New versions of browsers may "break" existing code. If you use a library then someone else will fix the problem - you can simply replace the library with the latest version.
- When you break the web browser rules; when you do "magic" you should draw attention to it with an animation.

on the other hand

- you import code that you do not need
- you may make simple thing complicated
- you make debugging rather harder

