PHP On-The-Fly!

This is an article on *PHP On-The-Fly!* in *PHP*.

PHP can be used for a lot of different things, and is one of the most powerful scripting languages available on the web. Not to mention it's extremely cheap and widely used. However, one thing that PHP is lacking, and in fact most scripting languages are, is a way to update pages in real-time, without having to reload a page or submit a form.

The internet wasn't made for this. The web browser closes the connection with the web server as soon as it has received all the data. This means that after this no more data can be exchanged. What if you want to do an update though? If you're building a PHP application (e.g. a high-quality content management system), then it'd be ideal if it worked almost like a native Windows/Linux application.

But that requires real-time updates. Something that isn't possible, or so you would think. A good example of an application that works in (almost) real-time is Google's GMail. Everything is JavaScript powered, and it's very powerful and dynamic. In fact, this is one of the biggest selling-points of GMail. What if you could have this in your own PHP websites as well? Guess what, I'm going to show you in this article.

How does it work?

If you want to execute a PHP script, you need to reload a page, submit a form, or something similar. Basically, a new connection to the server needs to be opened, and this means that the browser goes to a new page, losing the previous page. For a long while now, web developers have been using tricks to get around this, like using a 1x1 iframe, where a new PHP page is loaded, but this is far from ideal.

Now, there is a new way of executing a PHP script without having to reload the page. The basis behind this new way is a JavaScript component called the XML HTTP Request Object. Seehttp://jibbering.com/2002/4/httprequest.html for more information about the component. It is supported in all major browsers (Internet Explorer 5.5+, Safari, Mozilla/Firefox and Opera 7.6+).

With this object and some custom JavaScript functions, you can create some rather impressive PHP applications. Let's look at a first example, which dynamically updates the date/time.

Example 1

First, copy the code below and save it in a file called 'script.js':

Code:

```
var xmlhttp=false;
/*@cc_on @*/
/*@if (@_jscript_version >= 5)
// JScript gives us Conditional compilation, we can cope with old IE versions.
// and security blocked creation of the objects.
 try {
  xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
 } catch (e) {
  try {
   xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
  } catch (E) {
   xmlhttp = false;
  }
 }
@end @*/
if (!xmlhttp && typeof XMLHttpRequest!='undefined') {
  xmlhttp = new XMLHttpRequest();
}

function loadFragmentInToElement(fragment_url, element_id) {
    var element = document.getElementById(element_id);
    element.innerHTML = '<em>Loading ...</em>';
    xmlhttp.open("GET", fragment_url);
    xmlhttp.onreadystatechange = function() {
        if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {
            element.innerHTML = xmlhttp.responseText;
        }
    }
    xmlhttp.send(null);
}
```

Then copy the code below, and paste it in a file called 'server1.php':

PHP Code:

```
<?php
echo date("l dS of F Y h:i:s A");
?>
```

And finally, copy the code below, and paste it in a file called 'client1.php'. Please note though that you need to edit the line that says 'http://www.yourdomain.com/server1.php' to the correct location of server1.php on your server.

HTML Code:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Strict//EN">
<html>
<head>
<title>Example 1</title>
<script src="script.js" type="text/javascript"></script>

<script type="text/javascript">
        function updatedate() {

        loadFragmentInToElement('http://www.yourdomain.com/server1.php',
'currentdate');
        }

</script>
</head>

<body>
        The current date is    <span id="currentdate"><?php echo date("l dS of
F Y h:i:s A"); ?></span>.<br /><br />

        <input type="button" value="Update date" OnClick="updatedate();" />
</body>



</html>
```

Now go to http://www.yourdomain.com/client1.php and click on the button that says 'Update date'. The date will update, without the page having to be reloaded. This is done with the XML HTTP Request object. This example can also be viewed online athttp://www.phpit.net/demo/php%20on%2...ly/client1.php.

Example 2

Let's try a more advanced example. In the following example, the visitor can enter two numbers, and they are added up by PHP (and not by JavaScript). This shows the true power of PHP and the XML HTTP Request Object.

This example uses the same script.js as in the first example, so you don't need to create this again. First, copy the code below and paste it in a file called 'server2.php':

PHP Code:

```
<?php

// Get numbers
$num1 = intval($_GET['num1']);
$num2 = intval($_GET['num2']);

// Return answer
echo ($num1 + $num2);

?>
```

And then, copy the code below, and paste it in a file called 'client2.php'. Please note though that you need to edit the line that says 'http://www.yourdomain.com/server2.php' to the correct location of server2.php on your server.

HTML Code:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Strict//EN">
<html>
<head>
<title>Example 2</title>
<script src="script.js" type="text/javascript"></script>

<script type="text/javascript">
        function calc() {
                num1 = document.getElementById ('num1').value;
                num2 = document.getElementById ('num2').value;

                var element = document.getElementById('answer');
                xmlhttp.open("GET",
'http://www.yourdomain.com/server2.php?num1=' + num1 + '&num2=' + num2);
                xmlhttp.onreadystatechange = function() {
                        if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {
                                element.value = xmlhttp.responseText;
                        }
                }
            xmlhttp.send(null);
        }
</script>
</head>

<body>
        Use the below form to add up two numbers. The answer is calculated by
a PHP script, and <em>not</em> with JavaScript. What's the advantage to this?
You can execute server-side scripts (PHP) without having to refresh the
page.<br /><br />

        <input type="text" id="num1" size="3" /> + <input type="text"
id="num2" size="3" /> = <input type="text" id="answer" size="5" />

        <input type="button" value="Calculate!" OnClick="calc();" />
</body>


</html>
```

When you run this example, you can add up two numbers, using PHP and no reloading at all! If you can't get this example to work, then have a look onhttp://www.phpit.net/demo/php%20on%2...ly/client3.php to see the example online.
[SIZE=3]

[b]Any Disadvantages...?[/MARK]
There are only two real disadvantages to this system. First of all, anyone who has JavaScript turned off, or their browser doesn't support the XML HTTP Request Object will not be able to run it. This means you will have to make sure that there is a non-JavaScript version, or make sure all your visitors have JavaScript enabled (e.g. an Intranet application, where you can require JS).

Another disadvantage is the fact that it breaks bookmarks. People won't be able to bookmark your pages, if there is any dynamic content in there. But if you're creating a PHP application (and not a PHP website), then bookmarks are probably not very useful anyway.

Conclusion

As I've shown you, using two very simple examples, it is entirely possible to execute PHP scripts, without having to refresh the page. I suggest you read more about the XML HTTP Request Object (http://jibbering.com/2002/4/httprequest.html) and its capabilities.

The things you can do are limitless. For example, you could create an extremely neat paging system, that doesn't require reloading at all. Or you could create a GUI for your PHP application, which behaves exactly like Windows XP. Just think about it!

Be aware though that JavaScript must be enabled for this to work. Without JavaScript this will be completely useless. So make sure your visitors support JavaScript, or create a non-JavaScript version as well.

Source: http://www.go4expert.com/articles/php-fly-t212/