Tips for Twits » Blog Archive » HOWTO

JUNE 25, 2008

With the controllable jQuery Progress Bar, writing a form upload progress bar seems like a piece of cake now. Hypothetically, all we need is to create the bar, poll for the progress of the file upload, derive the new progress bar value (in percentage) and set it.

To do that you need to prepare the php script to do it. By default PHP cant report the progress of upload progress. However people smarter than me have already solved that problem. In 5 mins i've found 2 solutions: the Alternative PHP Cache (APC) method as well as the UploadProgress method. Both of them are PECL packages. Because i couldnt get APC to work on my server properly, i'll document the UploadProgress more in detail here...

Step 1: Install the uploadprogress package. Really simple just run the following command

```
pecl install uploadprogress
```

Once that is done, register the extension to your PHP with the following line in your php.ini

```
extension=uploadprogress.so
```

then restart your apache/httpd

Step 2: Create the form and your progress bar

```
<form id="uploadform" enctype="multipart/form-data" method="post">
<input id="progress_key" name="UPLOAD_IDENTIFIER" type="hidden" value=
<input id="ulfile" name="ulfile" type="file" />
<input type="submit" value="Upload" />
    <span id="uploadprogressbar" class="progressbar">0%</span>
</form>
```

this creates the form with a file field as well as a unique UPLOAD_IDENTIFIER hidden field that allows our script to check the progress of the form submission.

Step 3: Next the script itself to check the respond with the progress of the form submission. Lets call this file *uploadprogress.php*

```
header("Cache-Control: no-cache, must-revalidate");
header("Expires: Mon, 26 Jul 1997 05:00:00 GMT");

if (@$_GET['id']) {
    echo json_encode(uploadprogress_get_info($_GET['id']));
    exit();
}
```

The header no-cache declarations circumvents IE's cache of the response. Basically this form does nothing but respond with a json encoded string of the uploadprogress_get_info function. The id argument is the same one we used in the form. Think of it as a form-upload-process-id. A typical response looks like this:

```
{"time_start":"1214384364","time_last":"1214384366","speed_average":"2
```

the response encodes a good deal of data about the form submission. most importantly for us: *bytes_uploaded* and *bytes_total*

Step 3: use jQuery and a timer to keep polling the page and update the

progress bar value

```
var progress_key = '';

// this sets up the progress bar
$(document).ready(function() {
    $("#uploadprogressbar").progressBar();
});

// fades in the progress bar and starts polling the upload progress af
function beginUpload() {
    $("#uploadprogressbar").fadeIn();
    setTimeout("showUpload()", 1500);
}

// uses ajax to poll the uploadprogress.php page with the id
// deserializes the json string, and computes the percentage (integer)
// update the jQuery progress bar
// sets a timer for the next poll in 750ms
function showUpload() {
    $.get("uploadprogress.php?id=" + progress_key, function(data) {
        if (!data)
            return;

        var response;
        eval ("response = " + data);

        if (!response)
            return;

        var percentage = Math.floor(100 * parseInt(response['bytes_up
        $("#uploadprogressbar").progressBar(percentage);

    });
    setTimeout("showUpload()", 750);
```

```
    setTimeout("showUpload()", 750);
}
```

viola! read the comments if you dont understand the code. it is _THAT_ straightforward. Of course there can be many improvements such as stopping the script when the upload reaches 100% but thats probably not really needed since the whole page is refreshed. But this approach allows the flexibility of ajax submissions and what nots.

Again, download the jQuery progressbar here: jQuery progressbar or view the demo here

Note: for the APC solution seekers out there, assuming you can get APC to work, the solution is not so different. A few changes will get your there:

- Change the HTML hidden form field name from *UPLOAD_IDENTIFIER* to *APC_UPLOAD_PROGRESS*
- Change the PHP *uploadprogress_get_info($_GET['id'])* to *apc_fetch('upload_'.$_GET['id']));*
- Change the Javascript percentage calculation from:
  *Math.floor(100 * parseInt(response['bytes_uploaded']) / parseInt(response['bytes_total']));*
  to:
  *Math.floor(100 * parseInt(response['current']) / parseInt(response['total']));*