

# **Page Excerpts Using CURL**

by  
Peter Lavin

January 25, 2005

## ***Overview***

The significance of links to external pages can be lost if those pages have a lot of clutter. If your server supports PHP, this article will show you how to extract and present only specific elements from an external page using the CURL package. No knowledge of PHP is required but the reader should be familiar with HTML.

## ***Introduction***

CURL is a library of functions that allows you to connect to different servers using a variety of protocols. In this article, we will only be concerned with the “http” protocol.

Most web servers have the CURL package installed but you can verify if yours does by creating a file called “info.php” with the following contents:

```
<?php
    phpinfo();
?>
```

Upload this file to your server and then open it in your browser. You should see a page that provides information about the configuration of PHP on your server. Use the “find” feature of your browser and search for “CURL”. If CURL is enabled you are all set to go.

## ***What to Display***

Point your browser at the site you wish to excerpt and when the page has loaded right click a blank area of the screen. Choose the drop-down menu option that allows you to view the HTML source code and find the portion of the page that you wish to display.

Select 10 or 15 characters of text that begin the portion of the page you are interested in. Make sure that the text you select occurs only at this one location by searching the page for other occurrences. If it is not unique keep selecting text until it is. Save this text selection to a file called “start.txt”. Now do the same thing for the terminating text and save this as “end.txt” making sure that the last selected character terminates what you want to display.

You now have everything you need to display only a portion of a page from an external site.

## ***The Code***

Find below the required code:

```
<html>
<head>
<title>Local Events</title>
</head>
<body>
<div style="margin-left:30px">
```

```

<?php
$url = "http://www.thedomain/page.html";
//unique text to determine start goes here
$start = "yourstart";
//insert end text here
$end = "yourend";
//give credit to the originator
echo "Courtesy of <a href = \"\$url\">$url.</a><br /><br />";
$ch = curl_init();
curl_setopt ($ch, CURLOPT_URL, $url );
curl_setopt ($ch, CURLOPT_HEADER, 0);
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, 1);
$result = curl_exec ($ch) or die ("Couldn't connect to $url.");
curl_close ($ch);
$startposition = strpos($result,$start);
if($startposition > 0){
    $endposition = strpos($result,$end, $startposition);
    //add enough chars to include the tag
    $endposition += strlen($end);
    $length = $endposition-$startposition;
    $result = substr($result,$startposition,$length);
    echo $result;
}else
    echo "<center><h3>Not found - try again later.</h3></center>";
?>
</div>
</body>
</html>

```

In this code, the CURL package returns the contents of a web page into a string variable. We then search this variable for the portion we are interested in, copy it and then output it to the browser.

Pay special attention to the bolded items above. Substitute the page address you are interested in excerpting for “www.thedomain/page.html” making sure that you don’t delete or overwrite “http://”. Retrieve the “start.txt” and the “end.txt” files and replace “yourstart” and “yourend”, being sure to preserve the quotation marks. With these simple changes you can incorporate specific content from an external web site into your own site.

We’ve chosen to display this page extract in its own page but you may insert this excerpt into an existing page by simply copying the PHP code and ignoring the HTML. Keep the reference to the originating page – you want to give credit where credit is due.

## ***Conclusion***

Should the page that you are excerpting change you will have to adjust the code accordingly. The originating host may remove the page altogether so check regularly to be sure that it is still there.

If you know something about PHP have a look at the documentation of the CURL package. The “http” protocol is probably most commonly used with CURL but other protocols such as “ftp” and telnet are also supported. With a little imagination I’m sure you can find many more uses for this package.

## ***About the Author***

Peter Lavin runs a Web Design/Development firm in Toronto, Canada. For more information visit <http://www.softcoded.com>. He may be reached at [peterlavin@sympatico.ca](mailto:peterlavin@sympatico.ca).