Using custom functions

Last Updated Tuesday, 01 November 2005

Using a custom function is as easy as eating cheese cake. They can be used like the builtin tags tmpl, sub, var, etc.

```
01Â Â Â Â Â Â Â  <patTemplate:tmpl name="page">
02Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  <div>
03Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  Today is
04Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  <patTemplate:time format="m/d/Y"/>
05Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  </div>
06Â Â Â Â Â Â Â  </patTemplate:tmpl>
```

This will output Today is [current date]., where [current day] will be replaced with the current date.

```
01Â Â Â  <patTemplate:tmpl name="root">
02Â Â Â Â Â Â Â Â Â  This is a template that is used to display code.
03Â Â Â Â Â Â Â Â Â  <patTemplate:phpHighlight><?PHP
04Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  $i = 0;
05Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  while( $i < 10 )
06Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  {
07Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  echo "This is line $i.<br />";
08Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  $i++;
09Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  }
10Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  ?>
11Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  </patTemplate:phpHighlight>
12Â Â Â  </patTemplate:tmpl>
```

In the above example, the PHP code enclosed between the <patTemplate:phpHighlight> tags will be syntax highlighted in the output.

Creating a custom Function

Custom Functions have to be placed in patTemplate/Function and extended from patTemplate_Function. You have to implement one method that represents the function:

string patTemplate_Function::call(array params, string content)

The reader will pass an associative array containing all attributes of your tag as well as a string as second parameter, which will contain all character data (and HTML tags) between the opening and closing tags.
In this method you may compute the result and return it as a string. This result will then be inserted in the template instead of your function tag as well as the enclosed content.

You may use all functions, that are located in the patTemplate/Function folder, as patTemplate will auto-load the classes.

Template functions will be evaluated while analyzing the template, there's currently no way to use template variables or any input from the PHP script in the funcitons. Functions may be nested, patTemplate will always evaluate the innermost function first.

Code examples

This example is the code used for the custom function time, which is able to display the current time as well as reformat any timestamp you pass to this.

```
01Â Â Â Â Â Â Â  <?PHP
02Â Â Â Â Â Â Â  /**
03Â Â Â Â Â Â Â  * patTemplate function that calculates the current time
04Â Â Â Â Â Â Â  * or any other time and returns it in the specified format.
05Â Â Â Â Â Â Â  *
06Â Â Â Â Â Â Â  * @packageÂ Â Â Â Â Â  patTemplate
07Â Â Â Â Â Â Â  * @subpackageÂ Â Â  Functions
08Â Â Â Â Â Â Â  * @authorÂ Â Â Â Â Â Â  Stephan Schmidt <schst@php.net>
09Â Â Â Â Â Â Â  */
10Â Â Â Â Â Â Â  class patTemplate_Function_Time extends patTemplate_Function
11Â Â Â Â Â Â Â  {
12Â Â Â Â Â Â Â Â Â Â  /**
```

```
13Â Â Â Â Â Â Â Â Â Â Â  * name of the function
14Â Â Â Â Â Â Â Â Â Â Â  * @accessÂ Â Â  private
15Â Â Â Â Â Â Â Â Â Â Â  * @varÂ Â Â Â Â Â  string
16Â Â Â Â Â Â Â Â Â Â Â  */
17Â Â Â Â Â Â Â Â Â Â Â  var $_name = 'Time';
18Â Â Â Â Â Â Â
19Â Â Â Â Â Â Â Â Â Â Â  /**
20Â Â Â Â Â Â Â Â Â Â Â  * call the function
21Â Â Â Â Â Â Â Â Â Â Â  *
22Â Â Â Â Â Â Â Â Â Â Â  * @accessÂ Â Â  public
23Â Â Â Â Â Â Â Â Â Â Â  * @paramÂ Â Â Â  arrayÂ Â Â  parameters of the function (= attributes of the tag)
24Â Â Â Â Â Â Â Â Â Â Â  * @paramÂ Â Â Â  stringÂ Â  content of the tag
25Â Â Â Â Â Â Â Â Â Â Â  * @returnÂ Â Â  stringÂ Â  content to insert into the template
26Â Â Â Â Â Â Â Â Â Â Â  */
27Â Â Â Â Â Â Â Â Â Â Â  function call( $params, $content )
28Â Â Â Â Â Â Â Â Â Â Â  {
29Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  if( !empty( $content ) )
30Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  {
31Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  $params['time'] = $content;
32Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  }
33Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â
34Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  if( isset( $params['time'] ) )
35Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  {
36Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  $params['time'] = strtotime( $params['time'] );
37Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  }
38Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  else
39Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  {
40Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  $params['time'] = time();
41Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  }
42Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â
43Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  return date( $params['format'], $params['time'] );
44Â Â Â Â Â Â Â Â Â Â Â  }
45Â Â Â Â Â Â Â  }?>
```

The next code snippet is all that is needed for the phpHighlight function.

```
01Â Â Â Â Â Â Â  <?PHP
02Â Â Â Â Â Â Â  /**
03Â Â Â Â Â Â Â  * patTemplate function that highlights PHP code in your templates
04Â Â Â Â Â Â Â  *
05Â Â Â Â Â Â Â  * @packageÂ Â Â Â Â Â Â  patTemplate
06Â Â Â Â Â Â Â  * @subpackageÂ Â Â  Functions
07Â Â Â Â Â Â Â  * @authorÂ Â Â Â Â Â Â  Stephan Schmidt <schst@php.net>
08Â Â Â Â Â Â Â  */
09Â Â Â Â Â Â Â  class patTemplate_Function_Phphighlight extends patTemplate_Function
10Â Â Â Â Â Â Â  {
11Â Â Â Â Â Â Â Â Â Â Â  /**
12Â Â Â Â Â Â Â Â Â Â Â  * name of the function
13Â Â Â Â Â Â Â Â Â Â Â  * @accessÂ Â Â  private
14Â Â Â Â Â Â Â Â Â Â Â  * @varÂ Â Â Â Â Â  string
15Â Â Â Â Â Â Â Â Â Â Â  */
16Â Â Â Â Â Â Â Â Â Â Â  var $_name = 'Phphighlight';
17Â Â Â Â Â Â Â
18Â Â Â Â Â Â Â Â Â Â Â  /**
19Â Â Â Â Â Â Â Â Â Â Â  * call the function
20Â Â Â Â Â Â Â Â Â Â Â  *
21Â Â Â Â Â Â Â Â Â Â Â  * @accessÂ Â  public
22Â Â Â Â Â Â Â Â Â Â Â  * @paramÂ Â Â  arrayÂ Â Â  parameters of the function (= attributes of the tag)
23Â Â Â Â Â Â Â Â Â Â Â  * @paramÂ Â Â  stringÂ Â  content of the tag
24Â Â Â Â Â Â Â Â Â Â Â  * @returnÂ Â  stringÂ Â  content to insert into the template
25Â Â Â Â Â Â Â Â Â Â Â  */
26Â Â Â Â Â Â Â Â Â Â Â  function call( $params, $content )
27Â Â Â Â Â Â Â Â Â Â Â  {
28Â Â Â Â Â Â Â Â Â Â Â Â Â Â Â  ob_start();
```

```
Â Â Â Â Â Â Â Â Â Â Â Â Â Â  highlight_string( $content );
Â Â Â Â Â Â Â Â Â Â Â Â Â Â  $content = ob_get_contents();
Â Â Â Â Â Â Â Â Â Â Â Â Â Â  ob_end_clean();
Â Â Â Â Â Â Â Â Â Â Â Â Â Â  return $content;
Â Â Â Â Â Â Â Â Â Â Â  }
Â Â Â Â Â Â Â  }
Â Â Â Â Â Â Â  ?>
```