Legacy Mambots - The Old Way

Last Updated Wednesday, 14 September 2005

This example is for writing Mambo 4.5.0 mambots. You can still write mambots in this style for Mambo 4.5.1 and later providing they are saved in the /mambots directory. These mambots are loaded every time a content item is displayed. So, for the frontpage component, the same file could be loaded five, ten or twenty times depending on the number of content items displayed which is extremely inefficent. You are encouraged to look at upgrading your mambots to gain better performance and flexibility. Additionally, you should upgrade as this method will be deprecated in future versions (after Mambo 4.5.1 / Joomla 1.0).

For our development example, we'll imagine we are creating a Mambot to replace smile text with smile images.

The file for the mambot will be called mossmiles.php and it is saved in /mambots/mossmiles.php. It looks like this:

```
if (!defined( '_MOS_CONVERT_SMILES_MAMBOT' )) {
Â // only define the function once
Â function MAMBOT_convert_smiles( &$row ) {
Â  $smiles_src = array(
Â Â  ':)',
Â Â  ':('
Â  );
Â  $prefix = '<img src="images/smiles/';
Â  $suffix = ' height="12" width="12" alt="" />';
Â  $smiles_img = array(
Â Â  "{$prefix}happy.png{$suffix}",
Â Â  "{$prefix}sad.png{$suffix}" )
Â  );
Â  $row->text = str_replace( $smiles_src, $smiles_img, $row->text );
Â }
Â define( '_MOS_CONVERT_SMILES_MAMBOT', 1 );
}
MAMBOT_convert_smiles( $row );
```

A number of things are set up for a mambot. The mossmiles.php has access to a number of variables:

$rowthe current mosContent object. This is an object with all of the fields in the mos_content table.
$mosConfig_absolute_path

$maska variable holding masks for various display options

In addition, the $row object has a property called text that the mambot will operate on to make changes to the text.

Row
is passed by address to the MAMBOT_convert_smiles function so that any changes to the object made within the Mambot function are retained.

Because
a mambot is called more than once during the code execution, you must ensure that the function is defined only once. Hence, a constant is defined the first time the mambot is loaded. On subsequent load, the function definition is ignored.

All modifications to the content text are made to the $row->text property variable.