

Extention Writing

Vancouver PHP Conference

January 22nd, 2004. Vancouver, Canada

Derick Rethans <derick@php.net>

protos

```
return_type function_name(type1 param1, type2 param2 [, type3 param3])  
    Description
```

Types:

- o bool
- o int
- o string
- o float
- o array
- o object
- o resource

- o Go to the php5/ext directory
- o Execute the following command:

```
# ./ext_skel --extname=my_ext_name --proto=/path/to/protofile
```

- o This produces the following source files:
- o config.m4
- o my_ext_name.c
- o php_my_ext_name.h

config.m4

```

<html> <head> <title>config.m4</title> </head> <body bgcolor="#ffffff" text="#000000">
<pre> <a name="line4"> 4</a> PHP_ARG_WITH(my_ext_name, for my_ext_name support, <a
name="line5"> 5</a> [ --with-my_ext_name Include my_ext_name support]) <a
name="line6"> 6</a> <a name="line7"> 7</a> if test &quot;$PHP_MY_EXT_NAME&quot; !=
&quot;no&quot;; then <a name="line8"> 8</a> <a name="line9"> 9</a>
SEARCH_PATH=&quot;/usr/local /usr&quot;; <font color="#444444"># you might want to
change this</font> <a name="line10">10</a>
SEARCH_FOR=&quot;/<strong>include</strong>/my_ext_name.h&quot;; <font
color="#444444"># you most likely want to change this</font> <a name="line11">11</a> if test
-r $PHP_MY_EXT_NAME/$SEARCH_FOR; then <font color="#444444"># path given as
parameter</font> <a name="line12">12</a>
MY_EXT_NAME_DIR=$PHP_MY_EXT_NAME <a name="line13">13</a> else <font
color="#444444"># search default path list</font> <a name="line14">14</a>
AC_MSG_CHECKING([for my_ext_name files in default path]) <a name="line15">15</a> for
i in $SEARCH_PATH ; do <a name="line16">16</a> if test -r $i/$SEARCH_FOR; then <a
name="line17">17</a> MY_EXT_NAME_DIR=$i <a name="line18">18</a>
AC_MSG_RESULT(found in $i) <a name="line19">19</a> fi <a name="line20">20</a>
done <a name="line21">21</a> fi <a name="line22">22</a> <a name="line23">23</a> if test
-z &quot;$MY_EXT_NAME_DIR&quot;; then <a name="line24">24</a>
AC_MSG_RESULT([not found]) <a name="line25">25</a> AC_MSG_ERROR([Please
reinstall the my_ext_name distribution]) <a name="line26">26</a> fi </pre> </body> </html>

```

config.m4

```

<pre>
    <a name="line28">28</a>
PHP_ADD_INCLUDE($MY_EXT_NAME_DIR/<strong>include</strong>) <a
name="line29">29</a> <a name="line30">30</a> LIBNAME=my_ext_name <font
color="#444444"># you may want to change this</font> <a name="line31">31</a>
LIBSYMBOL=aa_init <font color="#444444"># you most likely want to change this </font> <a
name="line32">32</a> <a name="line33">33</a>
PHP_CHECK_LIBRARY($LIBNAME,$LIBSYMBOL, <a name="line34">34</a> [ <a
name="line35">35</a> PHP_ADD_LIBRARY_WITH_PATH($LIBNAME,
$MY_EXT_NAME_DIR/lib, MY_EXT_NAME_SHARED_LIBADD) <a
name="line36">36</a> AC_DEFINE(HAVE_MY_EXT_NAMELIB,1,[ ] <a
name="line37">37</a> ],[ <a name="line38">38</a> AC_MSG_ERROR([wrong
my_ext_name lib version or lib not found]) <a name="line39">39</a> ],[ <a
name="line40">40</a> -L$MY_EXT_NAME_DIR/lib -lm -ldl <a name="line41">41</a> ] )
<a name="line42">42</a> <a name="line43">43</a>
PHP_SUBST(MY_EXT_NAME_SHARED_LIBADD) <a name="line44">44</a> <a
name="line45">45</a> PHP_NEW_EXTENSION(my_ext_name, my_ext_name.c, $ext_shared)
<a name="line46">46</a> fi </pre>

```

- o Copy our newly created extension to some other place:

```
# mv ext/my_ext_name /dat/dev/php
```

- o Go to the new directory and run phpize:

```
# cd /dat/dev/php/my_ext_name  
# phpize
```

- o You should see the following output (or similar):

```
Configuring for:  
PHP Api Version:      20031224  
Zend Module Api No:   20020429  
Zend Extension Api No: 90021012
```

Run configure to test for the library and create the Makefile:

```
# ./configure
creating cache ./config.cache
checking for gcc... gcc
checking whether the C compiler (gcc ) works... yes
...
creating ./config.status
creating config.h
```

Now run make:

```
# make
```

And of course we get lots of compile errors :-)

- o Includes
- o Function table
- o Module entry
- o INI settings
- o Module/Request start and shutdown handlers
- o phpinfo() function
- o Function implementations

aalib.c:

```

<pre> <a name="line117">117</a> <font color="#444444"> /* {{{ proto resource aa_init() <a
name="line118">118</a>  Initialize the AA library */</font> <a name="line119">119</a> <font
color="#2040a0">PHP_FUNCTION</font><font color="4444FF"></font><font
color="#2040a0">aa_init</font><font color="4444FF"></font> <a name="line120">120</a>
<font color="4444FF"><strong>{</strong></font> <a name="line121">121</a> <font
color="#ff0000">aa_context *c;</font> <a name="line122">122</a> <a
name="line123">123</a> <strong>if</strong> <font color="4444FF"></font><font
color="#2040a0">ZEND_NUM_ARGS</font><font color="4444FF"></font><font
color="4444FF"></font> <font color="4444FF">!</font><font color="4444FF">=</font> <font
color="#FF0000">0</font><font color="4444FF"></font> <font
color="4444FF"><strong>{</strong></font> <a name="line124">124</a> <font
color="#2040a0">WRONG_PARAM_COUNT</font><font color="4444FF"></font> <a
name="line125">125</a> <font color="4444FF"><strong>}</strong></font> <a
name="line126">126</a> <a name="line127">127</a> <font color="#ff0000">c =
aa_init(&amp;mem_d, &amp;aa_defparams, NULL);</font> <a name="line128">128</a> <font
color="#ff0000">ZEND_REGISTER_RESOURCE(return_value, c, le_aalib);</font> <a
name="line129">129</a> <font color="4444FF"><strong>}</strong></font> <a
name="line130">130</a> <font color="#444444">/ } } /</font> </pre>

```

aalib.c:

```

<pre> <a name="line77"> 77</a> <strong>static</strong> <strong>void</strong> <font
color="#2040a0">rsrc_close_aalib</font><font color="4444FF">(</font><font
color="#2040a0">zend_rsrc_list_entry</font> <font color="4444FF">*</font><font
color="#2040a0">rsrc</font> <font color="#2040a0">TSRMLS_DC</font><font
color="4444FF">)</font> <a name="line78"> 78</a> <font
color="4444FF"><strong>{</strong></font> <a name="line79"> 79</a> <font
color="#2040a0">aa_context</font> <font color="4444FF"></font><font
color="#2040a0">c</font> <font color="4444FF">=</font> <font color="4444FF">(</font><font
color="#2040a0">aa_context</font> <font color="4444FF"></font><font
color="4444FF">)</font> <font color="#2040a0">rsrc</font><font
color="4444FF">-</font><font color="4444FF">&gt;</font><font
color="#2040a0">ptr</font><font color="4444FF">;</font> <a name="line80"> 80</a> <font
color="#2040a0">aa_close</font><font color="4444FF">)</font><font
color="#2040a0">c</font><font color="4444FF">)</font><font color="4444FF">);</font><font
name="line81"> 81</a> <font color="4444FF"><strong>}</strong></font> <a name="line82">
82</a> <a name="line83"> 83</a> <a name="line84"> 84</a> <font
color="#2040a0">PHP_MINIT_FUNCTION</font><font color="4444FF">(</font><font
color="#2040a0">aalib</font><font color="4444FF">)</font> <a name="line85"> 85</a> <font
color="4444FF"><strong>{</strong></font> <a name="line86"> 86</a> <font
color="#2040a0">le_aalib</font> <font color="4444FF">=</font> <font
color="#2040a0">zend_register_list_destructors_ex</font><font color="4444FF">(</font><font
color="#2040a0">rsrc_close_aalib</font>, <font color="#2040a0">NULL</font>, <font
color="#008000">&quot;aa&quot;</font>, <font color="#2040a0">module_number</font><font
color="4444FF">)</font><font color="4444FF">;</font> <a name="line87"> 87</a> <font
color="4444FF"><strong>}</strong></font> </pre>

```

aalib.c:

```

<pre>  <a name="line132">132</a>  <font color="#444444">/*  {{{  proto  int
aa_get_screen_height(resource c) <a name="line133">133</a>  Get the screen height */</font>
<a name="line134">134</a>  <font color="#2040a0">PHP_FUNCTION</font><font
color="4444FF"></font><font color="#2040a0">aa_get_screen_height</font><font
color="4444FF"></font>  <a name="line135">135</a>  <font
color="4444FF"><strong>{</strong></font>  <a name="line136">136</a>  <strong>int</strong>
<font color="#2040a0">argc</font>  <font color="4444FF">=</font>  <font
color="4444FF"></font><font color="4444FF"></font>  <a name="line137">137</a>
<strong>int</strong> <font color="#2040a0">c_id</font> <font color="4444FF">=</font> <font
color="4444FF"></font><font color="#FF0000">1</font><font color="4444FF">;</font> <a
name="line138">138</a>  <font color="#2040a0">zval</font>  <font
color="4444FF">*</font><font color="#2040a0">c</font> <font color="4444FF">=</font>
<font color="#2040a0">NULL</font><font color="4444FF">;</font>  <a
name="line139">139</a>  <font color="#ff0000">aa_context  *ac;</font>  <a
name="line140">140</a>  <a name="line141">141</a>  <strong>if</strong>  <font
color="4444FF"></font><font color="#2040a0">zend_parse_parameters</font><font
color="4444FF"></font><font color="#2040a0">argc</font>  <font
color="#2040a0">TSRMLS_CC</font>, <font color="#008000">&quot;r&quot;</font>, <font
color="4444FF">&amp;</font><font color="#2040a0">c</font><font color="4444FF"></font>
<font color="4444FF">=</font><font color="4444FF">=</font>  <font
color="#2040a0">FAILURE</font><font color="4444FF"></font>  <a
name="line142">142</a>  <strong>return</strong><font color="4444FF">;</font>  <a
name="line143">143</a>  <a name="line144">144</a>  <strong>if</strong>  <font
color="4444FF"></font><font color="#2040a0">c</font><font color="4444FF"></font>  <font
color="4444FF"><strong>{</strong></font>  <a name="line145">145</a>  <font
color="#ff0000">ZEND_FETCH_RESOURCE(ac, aa_context *, &amp;c, c_id, &quot;aa&quot;,
le_aalib);</font>  <a name="line146">146</a>  <font
color="4444FF"><strong>}</strong></font>  <a name="line147">147</a>  <font
color="4444FF"><strong>}</strong></font>  <a name="line148">148</a>  <font
color="#444444">}</font>  </pre>

```

Run make:

```
# make
```

```
...
```

```
Libraries have been installed in:  
  /dat/dev/php/my_ext_name/modules
```

Test library:

```
# php -r 'dl("/dat/dev/php/my_ext_name/modules/my_ext_name.so");  
          echo "\n".confirm_my_ext_name_compiled("my_ext_name")."\n";'
```

Output:

```
Congratulations! You have successfully modified ext/my_ext_name/config.m4.  
Module my_ext_name is now compiled into PHP.
```



```
Optional long and string:<pre> <strong>long</strong> <font color="#2040a0">opt_l</font>
<font color="4444FF">=</font> <font color="#FF0000">0</font><font
color="4444FF">;</font> <strong>char</strong> <font color="4444FF">*</font><font
color="#2040a0">opt_str</font> <font color="4444FF">=</font> <font
color="#2040a0">NULL</font><font color="4444FF">;</font> <strong>long</strong> <font
color="#2040a0">opt_str_len</font> <font color="4444FF">=</font> <font
color="#FF0000">0</font><font color="4444FF">;</font> <strong>if</strong> <font
color="4444FF"></font><font color="#2040a0">zend_parse_parameters</font><font
color="4444FF"></font><font color="#2040a0">ZEND_NUM_ARGS</font><font
color="4444FF"></font><font color="4444FF"></font> <font
color="#2040a0">TSRMLS_CC</font>, <font color="#008000">&quot;|s&quot;</font>,
<font color="4444FF">&amp;</font><font color="#2040a0">opt_l</font>, <font
color="4444FF">&amp;</font><font color="#2040a0">opt_str</font>, <font
color="4444FF">&amp;</font><font color="#2040a0">opt_str_len</font><font
color="4444FF"></font> <font color="4444FF">=</font><font color="4444FF">=</font> <font
color="#2040a0">FAILURE</font><font color="4444FF"></font> <font
color="4444FF"><strong>{</strong></font> </pre>
```

```
Get the first two parameters as resource and optional boolean:<pre> <font
color="#2040a0">zval</font> <font color="4444FF">*</font><font
color="#2040a0">resource</font><font color="4444FF">;</font> <font
color="#2040a0">zend_bool</font> <font color="#2040a0">b</font> <font
color="4444FF">=</font> <font color="#2040a0">FALSE</font><font
color="4444FF">;</font> <strong>if</strong> <font color="4444FF"></font><font
color="#2040a0">zend_parse_parameters</font><font color="4444FF"></font><font
color="#FF0000">2</font> <font color="#2040a0">TSRMLS_CC</font>, <font
color="#008000">&quot;r|b&quot;</font>, <font color="4444FF">&amp;</font><font
color="#2040a0">resource</font>, <font color="4444FF">&amp;</font><font
color="#2040a0">b</font><font color="4444FF"></font> <font color="4444FF">=</font><font
color="4444FF">=</font> <font color="#2040a0">FAILURE</font><font
color="4444FF"></font> <font color="4444FF"><strong>{</strong></font> </pre>
```

```
Get a long or a string:<pre> <strong>long</strong> <font color="#2040a0">number</font><font
color="4444FF">;</font> <strong>char</strong> <font color="4444FF">*</font><font
color="#2040a0">str</font><font color="4444FF">;</font> <strong>int</strong> <font
color="#2040a0">str_len</font><font color="4444FF">;</font> <strong>if</strong> <font
color="4444FF"></font><font color="#2040a0">zend_parse_parameters_ex</font><font
color="4444FF"></font><font color="#2040a0">ZEND_PARSE_PARAMS_QUIET</font>,
<font color="#2040a0">ZEND_NUM_ARGS</font><font color="4444FF"></font><font
color="4444FF"></font> <font color="#2040a0">TSRMLS_CC</font>, <font
color="#008000">&quot;l&quot;</font>, <font color="4444FF">&amp;</font><font
color="#2040a0">number</font><font color="4444FF"></font> <font
color="4444FF">=</font><font color="4444FF">=</font> <font
color="#2040a0">SUCCESS</font><font color="4444FF"></font> <font
color="4444FF"><strong>{</strong></font> <font color="#444444">/ Do something with the
number /</font> <font color="4444FF"><strong>}</strong></font> <strong>else</strong>
<strong>if</strong> <font color="4444FF"></font><font
color="#2040a0">zend_parse_parameters_ex</font><font color="4444FF"></font><font
color="#2040a0">ZEND_PARSE_PARAMS_QUIET</font>, <font
color="#2040a0">ZEND_NUM_ARGS</font><font color="4444FF"></font><font
color="4444FF"></font> <font color="#2040a0">TSRMLS_CC</font>, <font
color="#008000">&quot;s&quot;</font>, <font color="4444FF">&amp;</font><font
color="#2040a0">str</font>, <font color="4444FF">&amp;</font><font
color="#2040a0">str_len</font><font color="4444FF"></font> <font
color="4444FF">=</font><font color="4444FF">=</font> <font
color="#2040a0">SUCCESS</font><font color="4444FF"></font> <font
color="4444FF"><strong>{</strong></font> <font color="#444444">/ Do something with the
string /</font> <font color="4444FF"><strong>}</strong></font> <strong>else</strong> <font
color="4444FF"><strong>{</strong></font> <font color="#444444">/ Output error /</font>
<font color="4444FF"><strong>}</strong></font> </pre>
```

Returning from a function

```

<pre> <a name="line1"> 1</a> <font color="#2040a0">RETURN_NULL</font><font
color="4444FF"></font><font color="4444FF"></font> <a name="line3"> 3</a> <font
color="#2040a0">zend_bool</font> <font color="#2040a0">b</font><font
color="4444FF"></font> <a name="line4"> 4</a> <font
color="#2040a0">RETURN_BOOL</font><font color="4444FF"></font><font
color="#2040a0">b</font><font color="4444FF"></font> <a name="line5"> 5</a> <font
color="#2040a0">RETURN_FALSE</font> <a name="line6"> 6</a> <font
color="#2040a0">RETURN_TRUE</font> <a name="line8"> 8</a> <strong>long</strong>
<font color="#2040a0">l</font><font color="4444FF"></font> <a name="line9"> 9</a> <font
color="#2040a0">RETURN_LONG</font><font color="4444FF"></font><font
color="#2040a0">l</font><font color="4444FF"></font> <a name="line11">11</a>
<strong>double</strong> <font color="#2040a0">d</font><font color="4444FF"></font> <a
name="line12">12</a> <font color="#2040a0">RETURN_DOUBLE</font><font
color="4444FF"></font><font color="#2040a0">d</font><font color="4444FF"></font> <a
name="line14">14</a> <strong>char</strong> <font color="4444FF">*</font><font
color="#2040a0">s</font><font color="4444FF"></font> <a name="line15">15</a>
<strong>int</strong> <font color="#2040a0">l</font><font color="4444FF"></font> <a
name="line16">16</a> <strong>int</strong> <font color="#2040a0">duplicate</font><font
color="4444FF"></font> <a name="line17">17</a> <font
color="#2040a0">RETURN_STRING</font><font color="4444FF"></font><font
color="#2040a0">s</font>, <font color="#2040a0">duplicate</font><font
color="4444FF"></font> <a name="line18">18</a> <font
color="#2040a0">RETURN_STRINGL</font><font color="4444FF"></font><font
color="#2040a0">s</font>, <font color="#2040a0">l</font>, <font
color="#2040a0">duplicate</font><font color="4444FF"></font> <a name="line19">19</a>
<font color="4444FF"></font><font color="4444FF"></font> <a name="line21">21</a> <font
color="#2040a0">zval</font> <font color="4444FF">*</font><font
color="#2040a0">r</font><font color="4444FF"></font> <a name="line22">22</a> <font
color="#2040a0">RETURN_RESOURCE</font><font color="4444FF"></font><font
color="#2040a0">r</font><font color="4444FF"></font> </pre>

```

Returning an array

```

<pre> <a name="line253">253</a> <font color="#2040a0">PHP_FUNCTION</font><font
color="4444FF">(</font><font
color="4444FF">(</font><font
color="4444FF">)</font> <a
name="line254">254</a> <font
color="4444FF"><strong>{</strong></font> <a name="line255">255</a> <strong>int</strong>
<font color="#2040a0">argc</font> <font color="4444FF">=</font> <font
color="#2040a0">ZEND_NUM_ARGS</font><font color="4444FF">(</font><font
color="4444FF">)</font><font color="4444FF">;</font> <a name="line256">256</a>
<strong>int</strong> <font color="#2040a0">c_id</font> <font color="4444FF">=</font> <font
color="4444FF">-</font><font color="#FF0000">1</font><font color="4444FF">;</font> <a
name="line257">257</a> <font color="#2040a0">zval</font> <font
color="4444FF">*</font><font color="#2040a0">c</font> <font color="4444FF">=</font>
<font color="#2040a0">NULL</font><font color="4444FF">;</font> <a
name="line258">258</a> <font color="#2040a0">aa_context</font> <font
color="4444FF">*</font><font color="#2040a0">ac</font><font color="4444FF">;</font> <a
name="line259">259</a> <strong>char</strong> <font color="4444FF">*</font><font
color="#2040a0">buffer</font><font color="4444FF">;</font> <a name="line260">260</a>
<strong>int</strong> <font color="#2040a0">i</font><font color="4444FF">;</font> <a
name="line261">261</a> <a name="line262">262</a> <strong>if</strong> <font
color="4444FF">(</font><font color="#2040a0">zend_parse_parameters</font><font
color="4444FF">(</font><font color="#2040a0">argc</font>
color="#2040a0">TSRMLS_CC</font>, <font color="#008000">&quot;r&quot;</font>, <font
color="4444FF">&amp;</font><font color="#2040a0">c</font><font color="4444FF">)</font>
<font color="#2040a0">FAILURE</font><font color="4444FF">)</font> <a
name="line263">263</a> <strong>return</strong><font color="4444FF">;</font> <a
name="line264">264</a> <a name="line265">265</a> <strong>if</strong> <font
color="4444FF">(</font><font color="#2040a0">c</font><font color="4444FF">)</font> <font
color="4444FF"><strong>{</strong></font> <a name="line266">266</a> <font
color="#2040a0">ZEND_FETCH_RESOURCE</font><font color="4444FF">(</font><font
color="#2040a0">ac</font>, <font color="#2040a0">aa_context</font> <font
color="4444FF">*</font>, <font color="4444FF">&amp;</font><font
color="#2040a0">c</font>, <font color="#2040a0">c_id</font>, <font
color="#008000">&quot;aa&quot;</font>, <font color="#2040a0">le_aalib</font><font
color="4444FF">)</font><font color="4444FF">;</font> <a name="line267">267</a> <font
color="4444FF"><strong>}</strong></font> <a name="line268">268</a> <a
name="line269">269</a> <strong>if</strong> <font color="4444FF">(</font><font
color="#2040a0">array_init</font><font color="4444FF">(</font><font
color="#2040a0">return_value</font><font color="4444FF">)</font> <font
color="4444FF">=</font><font color="4444FF">=</font> <font
color="#2040a0">FAILURE</font><font color="4444FF">)</font> <font
color="4444FF"><strong>{</strong></font> <a name="line270">270</a> <font
color="#2040a0">RETURN_FALSE</font><font color="4444FF">;</font> <a
name="line271">271</a> <font color="4444FF"><strong>}</strong></font> <a
name="line272">272</a> <a name="line273">273</a> <font color="#2040a0">buffer</font>
<font color="4444FF">=</font> <font color="#2040a0">ecalloc</font><font
color="4444FF">(</font><font color="#2040a0">aa_scrheight</font><font
color="4444FF">(</font><font color="#2040a0">ac</font><font color="4444FF">)</font> <font
color="4444FF">*</font> <font color="#2040a0">aa_scrwidth</font><font
color="4444FF">(</font><font color="#2040a0">ac</font><font color="4444FF">)</font> <font
color="4444FF">+</font> <font color="#FF0000">1</font>, <font
color="#FF0000">1</font><font color="4444FF">)</font><font color="4444FF">;</font> <a
name="line274">274</a> <font color="#2040a0">memcpy</font><font
color="4444FF">(</font><font color="#2040a0">buffer</font>, <font
color="4444FF">(</font><font color="#2040a0">aa_text</font><font
color="4444FF">(</font><font color="4444FF">)</font>, <font
color="#2040a0">aa_scrheight</font><font color="4444FF">(</font><font
color="#2040a0">ac</font><font color="4444FF">)</font> <font color="4444FF">*</font>
<font color="#2040a0">aa_scrwidth</font><font color="4444FF">(</font><font
color="#2040a0">ac</font><font color="4444FF">)</font><font color="4444FF">)</font><font
color="4444FF">;</font> <a name="line275">275</a> <a name="line276">276</a>
<strong>for</strong> <font color="4444FF">(</font><font color="#2040a0">i</font> <font
color="4444FF">=</font> <font color="#FF0000">0</font><font color="4444FF">:</font>

```


Returning a 2D array

```

<pre> <a name="line285">285</a> <font color="#2040a0">PHP_FUNCTION</font><font
color="4444FF"></font><font color="#2040a0">aa_get_array</font><font
color="4444FF"></font> <a name="line286">286</a> <font
color="4444FF"></font><strong>{</strong></font> <a name="line287">287</a>
<strong>int</strong> <font color="#2040a0">argc</font> <font color="4444FF">=</font> <font
color="#2040a0">ZEND_NUM_ARGS</font><font color="4444FF"></font><font
color="4444FF"></font><font color="4444FF">;</font> <a name="line288">288</a>
<strong>int</strong> <font color="#2040a0">c_id</font> <font color="4444FF">=</font> <font
color="4444FF"></font><font color="#FF0000">1</font><font color="4444FF">;</font> <a
name="line289">289</a> <font color="#2040a0">zval</font> <font
color="4444FF">*</font><font color="#2040a0">c</font> <font color="4444FF">=</font>
<font color="#2040a0">NULL</font><font color="4444FF">;</font> <a
name="line290">290</a> <font color="#2040a0">aa_context</font> <font
color="4444FF">*</font><font color="#2040a0">ac</font><font color="4444FF">;</font> <a
name="line291">291</a> <strong>char</strong> <font color="4444FF">*</font><font
color="#2040a0">buffer</font><font color="4444FF">;</font> <a name="line292">292</a>
<font color="#2040a0">zval</font> <font color="4444FF">*</font><font
color="#2040a0">line</font><font color="4444FF">;</font> <a name="line293">293</a>
<strong>int</strong> <font color="#2040a0">i</font>, <font color="#2040a0">j</font><font
color="4444FF">;</font> <a name="line302">302</a> <strong>if</strong> <font
color="#2040a0">array_init</font><font color="#2040a0">return_value</font><font
color="4444FF">=</font><font color="4444FF">=</font> <font color="4444FF"></font> <font
color="#2040a0">FAILURE</font><font color="4444FF"></font> <font
color="4444FF"><strong>{</strong></font> <a name="line303">303</a> <font
color="#2040a0">RETURN_FALSE</font><font color="4444FF">;</font> <a
name="line304">304</a> <font color="4444FF"><strong>}</strong></font> <a
name="line305">305</a> <a name="line306">306</a> <font
color="#2040a0">buffer</font> <font color="4444FF">=</font> <font
color="#2040a0">ecalloc</font><font color="4444FF"></font><font
color="4444FF"></font><font color="4444FF"></font><font color="4444FF">*</font>
<font color="#2040a0">aa_scrheight</font><font color="4444FF"></font>
<font color="#2040a0">ac</font><font color="4444FF"></font> <font color="4444FF">*</font>
<font color="#2040a0">aa_scrwidth</font><font color="4444FF"></font> <font color="4444FF"></font>
<font color="#2040a0">ac</font><font color="4444FF"></font> <font color="4444FF">+</font>
<font color="#FF0000">1</font>, <font color="#FF0000">1</font><font
color="4444FF"></font><font color="4444FF">;</font> <a name="line307">307</a> <font
color="#2040a0">memcpy</font><font color="4444FF"></font><font
color="#2040a0">buffer</font>, <font color="#2040a0">aa_text</font><font
color="4444FF"></font><font color="#2040a0">ac</font><font color="4444FF"></font>,
<font color="#2040a0">aa_scrheight</font><font color="4444FF"></font><font
color="#2040a0">ac</font><font color="4444FF"></font> <font color="4444FF">*</font>
<font color="#2040a0">aa_scrwidth</font><font color="4444FF"></font><font
color="4444FF"></font><font color="4444FF"></font><font color="4444FF"></font>
<font color="4444FF">;</font> <a name="line308">308</a> <a name="line309">309</a>
<strong>for</strong> <font color="4444FF"></font><font color="#2040a0">i</font> <font
color="4444FF">=</font> <font color="#FF0000">0</font><font color="4444FF">;</font>
<font color="#2040a0">i</font> <font color="4444FF">&lt;</font> <font
color="#2040a0">aa_scrheight</font><font color="4444FF"></font><font
color="#2040a0">ac</font><font color="4444FF"></font><font color="4444FF">;</font> <font
color="#2040a0">i</font><font color="4444FF">+</font><font color="4444FF">+</font><font
color="4444FF"></font> <font color="4444FF"><strong>{</strong></font> <a
name="line310">310</a> <font color="#2040a0">MAKE_STD_ZVAL</font><font
color="4444FF"></font><font color="#2040a0">line</font><font
color="4444FF"></font><font color="4444FF">;</font> <a name="line311">311</a>
<strong>if</strong> <font color="4444FF"></font><font color="4444FF"></font><font
color="#2040a0">array_init</font><font color="4444FF"></font><font
color="#2040a0">line</font><font color="4444FF"></font>
<font color="4444FF">=</font><font color="4444FF">=</font> <font
color="#2040a0">FAILURE</font><font color="4444FF"></font>
<font color="4444FF"><strong>{</strong></font> <a name="line312">312</a> <font
color="#2040a0">FREE_ZVAL</font><font color="4444FF"></font><font
color="#2040a0">line</font><font color="4444FF"></font><font color="4444FF">:</font> <a

```

?

Internals list: <mailto:internals@lists.php.net>

PECL list: <mailto:pecl-dev@lists.php.net>

These Slides: <http://pres.derickrethans.nl/encryption-vancouver>

PHP: <http://www.php.net>

Index

Prototypes	2
Framework generation	3
Search for the library	4
Check if the library works	5
First test compile	6
Running configure (2)	7
Extension structure	8
Resource creation	9
Resource destruction	10
Resource fetching	11
Compilation	12
Parameter Parsing API (1)	13
Parameter Parsing API (2)	14
Returning from a function	15
Returning an array	16
Returning a 2D array	17
Questions	18
Resources	19