

Re: trouble moving existing site having "self-contained" Pear packages defined within it

Re: trouble moving existing site having "self-contained" Pear packages defined within it

Source: <http://coding.derkeiler.com/Archive/PHP/alt.php/2007-03/msg00396.html>

- *From:* "Paul" <lof@xxxxxxxxxxxx>
 - *Date:* Wed, 28 Mar 2007 14:50:29 -0700
-

"Steve" <no.one@xxxxxxxxxxxx> wrote in message
[news:NJyOh.674\\$Zf5.147@xxxxxxxxxxxxxxxx](news:NJyOh.674$Zf5.147@xxxxxxxxxxxxxxxx)

"Paul" <lof@xxxxxxxxxxxx> wrote in message
[news:7WxOh.28213\\$68.27367@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](news:7WxOh.28213$68.27367@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)
| I am moving an existing app written years ago to a new server. It uses
| Sigma Template 1.3 and Quickform 1.1.1 and PEAR.php,v 1.1.1.1 2004/02/16
|
| The directory structure is like this:
| /site
| /site/html/Pear.php
| /site/html/Sigma.php
| /site/html/Common.php
| /site/html/Quickform.php
| /site/html/Quickform/
| /site/Mail/mail.php
| /pages/page1.php
| ...

you should create a site/application class and give it interfaces that
point
to paths. mine looks like this:

```
<?  
class application  
{  
    static private $_instance = null;  
  
    static public $adminEmail = "  
    static public $classDirectory = "  
    static public $cssDirectory = "  
    static public $currentPage = "  
    static public $description = "  
    static public $errorLogFile = "  
    static public $fontDirectory = "  
    static public $homePage = "  
    static public $host = "
```

Re: trouble moving existing site having "self-contained" Pear packages defined within it

```
static public $htdocsDirectory = "";
static public $imagesDirectory = "";
static public $includeDirectory = "";
static public $javascriptDirectory = "";
static public $lastSecurityCode = "";
static public $logo = "";
static public $mailDropDirectory = "";
static public $popupAttributes = "";
static public $rootDirectory = "";
static public $securityCode = "";
static public $title = "";
static public $uploadBaseDirectory = "";
static public $uri = "";

private function __clone(){ }

private function __construct(){ }

static function getInstance()
{
if (is_null(self::$_instance)){ self::$_instance = new application(); }
return self::$_instance;
}
}
?>
```

in your code, you should do something like:

```
$site = application::getInstance();
$site->includePath = '/site/inc/';
```

and in your other code, reference the interface:

```
require_once $site->includePath . 'some.script.php'
```

the site/application class should be put in on file that is required by *all* of your scripts at the beginning. when you roll out to another site (which could have a completely different directory structure), all you have

to do is edit this required script to point the class interfaces to the appropriate directories. i call mine 'site.cfg.php'. i also put this script

(relative.path.php) in php's include path:

```
<?
$parsedUri = dirname($_SERVER['PHP_SELF']);
$parsedUri .= substr($parsedUri, -1) != '/' ? '/' : '';
$relativeUri = str_replace('/', '', $parsedUri);
$relativePath = strlen($parsedUri) - strlen($relativeUri) - 1;
if ($relativePath < 0){ $relativePath = 0; }
$relativePath = str_repeat('../', $relativePath);
```

Re: trouble moving existing site having "self-contained" Pear packages defined within it

```
if (!$relativePath){ $relativePath = './'; }  
?>
```

so, every script i have starts out like this:

```
<?  
require_once 'relative.path.php';  
require_once $relativePath . 'site.cfg.php';  
?>
```

thus, it makes **no** difference how php is configured nor how deeply nested
a
script is in your directory structure, your script will always be pointing
to the correct files you require.

hth,

me

very cool idea – thanks