

---

## php configuration on aleph server

- **Article Type:** General
  - **Product:** Aleph
  - **Product Version:** 20
- 

### Description:

Has anyone configured php on aleph server with aleph v20?

We are trying to add a feature to our web catalogue where our users can sms the bib details of any item they have searched.

In order to get this working we need a php script in the backend which receives and forwards the users request (send via a js interface).

I have installed php (yum install) in the server. But I'm at loss as to which file I should configure for our server to understand php? In normal case I would have probably modified httpd.conf to load php module and then may be putting a file under standard doc root like /var/www/html might have worked.

But with Aleph the configuration of apache is not so straight forward. It does not look like modifying the httpd.conf at /etc/httpd..... works!

### Resolution:

[From Mathias Weyland, ETH Zurich, on the ALEPH500-DISCUSS-L:]

Aleph comes with its own apache located in \$alephe\_root/apache. If you modify /etc/httpd/httpd.conf, you are altering the configuration of the web server provided by your operating system, not the one installed by ExLibris. I guess there are several ways to enable php:

1. Run both apache servers, don't mess around with the one in \$alephe\_root/apache and set up php on the other one. (Of course you need to bind two different ports and you are probably wasting some resources.
2. Restrain from using the system web server and run your PHP in CGI mode.
3. Try to make ExLibris' web server load mod\_php5. I imagine this to be rather tricky since you probably can't just use any [libphp5.so](#) apache module because of ABI constraints, but I have no experience related to this third case. I personally wouldn't follow this path, though.

[Later:]

> As part of test I simply tried putting a test script in cgi-bin ( at  
> the default /var/www/cgi-bin)directory with .cgi extension instead of

> php. But then I guess it being a cgi script one won't be able to  
> execute it at command prompt to see if it works or not!

This should work if set up properly. What you need is a php binary. Your distribution probably comes with a package (or several packages) which provide(s) said binary. On my system, the php binary is located at /usr/bin/php5-cgi. Once you installed the package, you can run a dry test on the command line:

```
mweyland@cache:~$ cat test.php
<?php
print("foo\n");
?>
mweyland@cache:~$ /usr/bin/php5-cgi test.php
Content-type: text/html

foo
mweyland@cache:~$
```

Note that the HTTP content type header is provided. This is critical and you will get internal HTTP errors if php is set up to omit the headers. The output is what the web server is supposed to send to the browser once test.php is accessed. So we need to tell the web server that files with the extension .php and located in a specific directory have to be executed by /usr/bin/php5-cgi.

Once you have a working php interpreter, you can configure your apache. Locate the configuration file of the web server you want to enable for php scripts; that would be the one in \$alephe\_root/apache/conf if you go for step 2 of my last email. You need the following configuration statements:

```
ScriptAlias /php/ "/full/path/to/your/php/script/directory"
```

This links the URL /php/ to some directory which contains your php scripts. There probably already is such a ScriptAlias for cgi-bin. You can either stick with that one or add a dedicated ScriptAlias to separate your php scripts from the cgi-bin stuff ALEPH comes with.

```
AddType application/x-httpd-php .php
```

This links the extension .php to the action handler application/x-httpd-php.

```
Action application/x-httpd-php "/usr/bin/php5-cgi"
```

This links the action handler application/x-httpd-php to the interpreter /usr/bin/php5-cgi.

Also, make sure that you don't have any stale "LoadModule php..." entries from previous tests.

> and how do I  
> access it on browser (the std url would ofcourse point to online  
> catalogue as configured for aleph apache).

I am not quite sure whether I properly understood that part. Let's say you add

```
ScriptAlias /foo/ "/exlibris/aleph/php_project"
```

and save the php script bar.php in /exlibris/aleph/php\_project/, then the URL would be <http://your.host/foo/bar.php>

(You may need to specify the port if the web server does not run on port 80; the "left part" of the URL does not differ from the one you would use to access your WebOPAC. It's just that instead of /F?..., you access /foo/ or whatever you specify in your ScriptAlias directive.

<end of original response from Mathias Weyland>

[From site originally reporting the problem:]

I created two test files one as .php and one as .cgi

they have simple code like yours. The .php file has

```
#!/usr/bin/php
<?php
echo "test script";
?>
```

and .cgi file has :

```
#!/usr/bin/php-cgi
<?php
print "test script";
?>
```

and when I execute the .php I simply get back :

test script

and when I execute .cgi script I get back:

```
X-Powered-By: PHP/5.1.6
Content-type: text/html
```

test script

So it looks like my php/cgi is getting interpreted right(I suppose , hard coding the path to executables i.e. /usr/bin/php, is ok).

So looks like what I need to do next is to modify the apache config file in \$alephe\_root/apache/conf. I will try that and let you know how I get on.

[Response from Mathias Weyland:]

Yes indeed, this looks correct, I think you are half-way there. Maybe I was a bit too sloppy describing the apache configuration. Actually there are two different ways of setting this up. It all boils down to whether you need that first line in the script. (The so called magic number #! in '#!/usr/bin/php-cgi'). This line tells which program is to be used to execute the script. If you are able to place that magic number at the beginning of every php script of your project, you are good to go and do not

- 
- **Article last edited:** 10/8/2013