# PHP for VSI OpenVMS IA64 and Alpha

January 2021

*Important notes:*

- *Unlike the CSWS_PHP software kit provided by HPE, this PHP kit for VSI OpenVMS does not have any direct or implied association with CSWS (Apache HTTPD for OpenVMS) and it can be installed and used independently of CSWS. The kit does include both MOD_PHP5 (the PHP module for Apache HTTPD) and support for PHP CGI scripts with CSWS; however these optional facilities must be configured manually.*

- *This will be the final release of PHP 5.6.10 by VSI for VSI OpenVMS. This version is now some years old and latest versions provide numerous additional features. The primary motivation for this release is to introduce several additional extensions that were implemented some time ago but not previously released. Most notably, this release includes an extension for VSI DATATRIEVE that can be used to query and update indexed RMS files and other data sources supported by the DATATRIEVE product. These extensions will also be included in future releases of PHP for VSI OpenVMS.*

## 1. Introduction

Thank your for your interest in this port of PHP to OpenVMS. The current release of PHP for OpenVMS is based on the PHP 5.6.10 Open Source distribution.

PHP (http://php.net/) is a popular general-purpose Open Source scripting language that is especially suited to web development.

This OpenVMS port of PHP includes the core PHP language interpreter and various extensions (dynamically loadable modules), including extensions for various database client API's, XML, compression, arbitrary precision arithmetic, web services (SOAP and REST), and graphics. It is anticipated that additional extensions will be included in future releases.

## 2. What's new in this update

This update includes new dynamically loadable PHP modules that facilitate interaction with VSI DATATRIEVE, the Oracle RDB and Mimer relational databases, and the Redis in-memory data structure store (see https://redis.io). This release also includes a simple OpenVMS distributed lock manager extension that that can be used to coordinate access to shared resources, and there have been minor changes to the SQLite extension to improve error handling and reporting.

## 3. Requirements

The kit you are receiving has been compiled and built using the operating system and compiler versions listed below. While it is highly likely that you will have no problems installing and using the kit on systems running higher versions of the products listed, we cannot say for sure that you will be so lucky if your system is running older versions.

- VSI OpenVMS 8.4-1H1

- CSWS 2.4 for VSI OpenVMS (optional)

  CSWS 2.4 is required if you wish to use MOD_PHP5 to process PHP-based dynamic web pages. The `MOD_PHP5.EXE` shareable image provided with this kit is not compatible with earlier versions of CSWS.

- The software must be installed on an ODS-5 enabled disk (the installation will fail if this requirement is not met)

- HP TCP/IP Services V5.7 – ECO 5 (if you wish to perform network operations)

  It has not been verified whether the kit works with the MultiNet TCP/IP stack, but there is a good chance that it will.

In addition to the above requirements, it is assumed that the reader has a good knowledge of OpenVMS and of software development in the OpenVMS environment.

# 4. Before beginning the installation

Before installing PHP 5.6.10, you must manually remove any existing version of HP `CSWS_PHP` if it is installed on your system. Use PCSI to remove `CSWS_PHP` by entering the following command:

```
$ PRODUCT REMOVE CSWS_PHP
```

# 5. Installing the kit

The kit is provided as an OpenVMS PCSI kit (`VSI-I64VMS-PHP-V0506-10L-1.PCSI` or `VSI-AXPVMS-PHP-V0506-10L-1.PCSI`) that can be installed by a suitably privileged user using the following command:

```
$ PRODUCT INSTALL PHP
```

The installation will then proceed as follows (output may differ slightly from that shown):

```
Performing product kit validation of signed kits ...

The following product has been selected:
    VSI I64VMS PHP V5.6-10L              Layered Product [Installed]

Do you want to continue? [YES]

Configuration phase starting ...

You will be asked to choose options, if any, for each selected product and for
any products that may be installed to satisfy software dependency requirements.

Configuring VSI I64VMS PHP V5.6-10L: PHP for OpenVMS is based on PHP Version 5.6.10

    © Copyright 2017 VMS Software Inc.

    VSI Software Inc.

* This product does not have any configuration options.

Execution phase starting ...

The following product will be installed to destination:
    VSI I64VMS PHP V5.6-10L              DISK$I64SYS:[VMS$COMMON.]

Portion done: 0%...10%...20%...30%...40%...50%...90%...100%

The following product has been installed:
    VSI I64VMS PHP V5.6-10L              Layered Product
```

```
VSI I64VMS PHP V5.6-10L: PHP for OpenVMS is based on PHP Version 5.6.10

    Post-installation tasks are required.

    To start PHP at system boot time, add the following lines to
    SYS$MANAGER:SYSTARTUP_VMS.COM:

        $ file := SYS$STARTUP:PHP$STARTUP.COM
        $ if f$search("''file'") .nes. "" then @'file'

    To shutdown PHP at system shutdown time, add the following lines to
    SYS$MANAGER:SYSHUTDWN.COM:

        $ file := SYS$STARTUP:PHP$SHUTDOWN.COM
        $ if f$search("''file'") .nes. "" then @'file'
```

## 5.1.    Post-installation steps

After the installation has successfully completed, include the commands displayed at the end of the installation procedure into `SYSTARTUP_VMS.COM` to ensure that the logical names required in order for users to use the software are defined system-wide at start-up.

Users will then be able to use the PHP language interpreter by defining a foreign command as follows:

```
$ PHP :== $PHP$ROOT:[BIN]PHP.EXE
```

Generally speaking there are no special quota or privilege requirements required for users wishing to develop applications using PHP, although it should be noted that some extensions may have special requirements (for example, networking extensions may require a high `BYTLM` quota).

At this time you may also wish to customize the system-wide PHP initialization file (see `PHP$ROOT:[ETC]PHP.INI`) to define which PHP extensions should be automatically loaded by the interpreter and to specify any other default settings that may be relevant to your environment. Example initialization files for typical development and production use are provided in `PHP$ROOT:[ETC]`, and these may be used as the basis for the creation of a `PHP.INI` appropriate to your environment.

## 5.2.    Using MOD_PHP5.EXE with CSWS/Apache

In order to use PHP with CSWS 2.4 to process dynamic web pages using `MOD_PHP5`, the following steps must be performed. You may also wish to define one or more aliases for directories that will be used to serve PHP scripts.

1.  Copy `MOD_PHP5.EXE` into the CSWS modules directory

    ```
    $ COPY/LOG PHP$ROOT:[CSWS]MOD_PHP5.EXE APACHE$COMMON:[MODULES]
    ```

2.  Edit your CSWS configuration file (`APACHE$COMMON:[CONF]HTTPD.CONF`) and include the following directives. It is recommended that these directives be placed at the end of the file.

    ```
    ## Load PHP module
    LoadModule php5_module modules/mod_php5.exe

    ## Define types to be associated with MOD_PHP
    AddType application/x-httpd-php .php .phtml
    AddType application/x-httpd-php-source .phps
    ```

3. Stop and restart CSWS to pick up the modified configuration

```
$ @SYS$STARTUP:APACHE$SHUTDOWN.COM
$ @SYS$STARTUP:APACHE$STARTUP.COM
```

4. To test the PHP module, create a simple script named `DEMO.PHP` (or use any other name you wish) in `APACHE$COMMON:[HTDOCS]` with the following contents (or similar) and browse to http://my.host.name/demo.php, where *my.host.name* is the name or TCP/IP address of your web server.

```
<?php
echo "Testing the PHPINFO () function<br>\n";
phpinfo (INFO_ALL);
?>
```

If `MOD_PHP5` is working correctly, you should see displayed a web page that outlines the capabilities and setting of your PHP installation.

# 6.    Installing in an alternative location

By default the software will be installed in `SYS$SYSDEVICE:[VMS$COMMON]`. If you wish to install the software in an alternative location this can be achieved using the `/DESTINATION` qualifier with the `PRODUCT INSTALL` command to specify the desired location; however it is important to note that an additional manual step will then be required to complete the installation. Specifically, when an alternative destination is specified, the start-up and shutdown procedures (`PHP$STARTUP.COM` and `PHP$SHUTDOWN.COM`) will be placed into a subdirectory `[.SYS$STARTUP]` residing under the specified destination directory. If you wish to run these files from your standard `SYS$STARTUP` directory they will need to be copied from the destination subdirectory into your systems `SYS$STARTUP` directory.

# 7.    Known problems and restrictions

- Do not attempt use `MOD_PHP5.EXE` with versions of CSWS prior to version 2.4 (see previous comments).

- All PHP script files must be stream-lf.

- Composer Dependency Manager for PHP (https://getcomposer.org/) currently will not work correctly. It is anticipated that an OpenVMS-compatible version of this tool will be included in future releases of PHP for VSI OpenVMS.

- When using PHP with CSWS and `MOD_PHP`, note that `MOD_PHP` creates temporary files in `SYS$SCRATCH`, which for the `APACHE$WWW` account equates to `apache$root:[000000]`. These files are created with names of the form `ZendSem.XXXXXX`. To ensure that these files are deleted when web server worker processes terminate, define the logical name `DECC$ALLOW_REMOVE_OPEN_FILES` (to `"TRUE"`) in `APACHE$WWW`'s `LOGIN.COM`. Alternatively, put in place procedures to periodically purge these files

## 8. Included extensions

This release of PHP for OpenVMS includes the core PHP language and a number of commonly-used dynamically loadable extensions. The included extension modules are listed below.

- BCMATH.EXE
- BZ2.EXE
- CALENDAR.EXE
- CURL.EXE
- DBA.EXE
- EXIF.EXE
- FTP.EXE
- GD.EXE
- GMP.EXE
- LDAP.EXE
- MBSTRING.EXE
- MYSQL.EXE
- MYSQLI.EXE
- MYSQLND.EXE
- OCI8.EXE  (ia64 only)
- ODBC.EXE
- OPCACHE.EXE
- OPENSSL.EXE
- OPENVMS.EXE
- PDO_DBLIB_FREETDS.EXE
- PDO_MYSQL.EXE
- PDO_OCI.EXE  (ia64 only)
- READLINE.EXE
- SHMOP.EXE
- SOAP.EXE
- SOCKETS.EXE
- SYBASE_CT.EXE
- SYSVSEM.EXE
- SYSVSHM.EXE
- TIDY.EXE
- WDDX.EXE
- XMLRPC.EXE
- XSL.EXE
- ZIP.EXE

- `ZLIB.EXE`

- `ZMQ.EXE`

- `RDB.EXE`

- `MIMERDB.EXE` (ia64 only)

- `REDIS.EXE`

- `DTR.EXE`

- `DSCDEF.EXE`

- `REC.EXE`

Other commonly-used extensions are statically included in the PHP interpreter (the command "`PHP -m`" can be used to show details about these modules).