Article
Jon Willeke · Sep 20, 2016 3m read

Using ODBC with PHP in OS X 10.9 (Mavericks)

In OS X 10.8 and earlier, PHP was built with ODBC support, making it easy to connect to a Caché database:

```
$ php -r 'phpinfo();' |grep 'ODBC Support'
ODBC Support => enabled
```

However, this support was removed in OS X 10.9:

```
$ php -r 'phpinfo();' |grep 'ODBC Support'
```

Some people use Homebrew, MacPorts, or Fink to get their tools back. If you're comfortable coloring a little outside the lines (i.e., using sudo and the command line), you can build the odbc.so or pdoodbc.so extension (depending on which API you prefer), and use it with Apple's version of Apache and PHP. You'll need the following:

- C compiler
- iODBC headers
- GNU autoconf
- PHP source code

To get a C compiler, I installed the command line developer tools with a command like the following:

```
$ xcode-select --install
```

This opened a prompt that installed them to /Library/Developer. Alternatively, you can install Xcode from the Mac App Store.

OS X ships with the iODBC libraries in /usr/lib, but not the headers. I don't actually remember how I installed them, but I think I downloaded the source from iodbc.org, then copied the following header files to /usr/local/include:

- sql.h
- sqlext.h
- sqltypes.h
- sqlucode.h

The phpize utility used for building a PHP extension depends on GNU autoconf. I downloaded autconf 2.69 from a GNU mirror, then built and installed it to /usr/local using configure and make as follows:

```
$ ./configure && make && sudo make install
```

OS X 10.9 ships with PHP 5.4.45. Although you can download that version from a PHP mirror, I downloaded 5.6.26

to fix crashes in pdoodbc. I built odbc.so with the following commands in the PHP source directory:

```
$ cd ext/odbc
$ phpize
Configuring for:
PHP Api Version:         20100412
Zend Module Api No:      20100525
Zend Extension Api No:   220100525
$ CPPFLAGS=-DHAVE_IODBC ODBC_TYPE=iodbc ./configure --with-iodbc
...
config.status: creating config.h
$ make
...
Build complete.
```

The environment variables work around what appear to be broken sections of the configure script. At this point, I copied modules/odbc.so to /usr/lib/php/extensions/no-debug-non-zts-20100525. You can also sudo make install.

I built pdoodbc.so similarly:

```
$ cd ext/pdo_odbc
$ phpize
Configuring for:
PHP Api Version:         20100412
Zend Module Api No:      20100525
Zend Extension Api No:   220100525
$ ./configure --with-pdo-odbc=iODBC
...
config.status: creating config.h
$ make
...
Build complete.
```

With the extensions installed, I enabled them in php.ini. In the /etc directory, I renamed php.ini.default to php.ini, then added the following lines:

```
extension=pdo_odbc.so
extension=odbc.so
```

I had previously enabled the PHP module for Apache by editing httpd.conf in /etc/apache2. If you haven't already, uncomment this line:

```
LoadModule php5_module libexec/apache2/libphp5.so
```

Lastly, I restarted Apache:

```
$ sudo apachectl restart
```

Now I can develop with PHP and Caché locally on my Mac. Let me know if I've missed a step, or if something

works differently in later versions of OS X.

#ODBC #Caché