PHP Docker images with additional extensions

Ákos Takács

CONTENTS:

I started this project in 2016 to support PHP Docker images based on the official Docker images of PHP, but with the ability to add PHP extensions more easily without finding out every time which extension requires what libraries installed on the host or in this case in the container.

Since I obviously can't support different images for every possible combination of required extensions, I support one for each PHP version containing all the extensions supported by that version and this project.

You can still use this project to build your own image

ONE

SUPPORTED PHP VERSIONS AND TAGS

Although the official PHP Docker images have multiple variants, this project currently supports only Debian-based PHP FPM images.

Table 1: Supported tags

tags	Dockerfile
8.1-fpm, latest	8.1/fpm/Dockerfile
8.0-fpm	8.0/fpm/Dockerfile
7.4-fpm	7.4/fpm/Dockerfile
7.3-fpm	7.3/fpm/Dockerfile
7.2-fpm	7.2/fpm/Dockerfile
7.1-fpm	7.1/fpm/Dockerfile
7.0-fpm	7.0/fpm/Dockerfile
5.6-fpm	5.6/fpm/Dockerfile

You can use any of the above tags to pull a PHP Docker image. If you need PHP 8.1, just run

```
docker pull rimelek/php:8.1-fpm
```

You can also get more detauls about the images and tags without pulling them by going to Docker Hub: https://hub.docker.com/r/rimelek/php

Note: You might find some old references to `itsziget/php` which is the previous name of this image repository. I realize that I can't just move an image to a different repository and move it back, so I am going to keep that repository indefinitely as an alias of `rimelek/php`, but it is recommended to use the namespace `rimelek`, since this is what I will focus on and probably check more frequently.

ADDITIONAL EXTENSIONS

PHP (installed)

- bcmatch
- bz2
- calendar
- exif
- gd
- gettext
- gmp
- imap
- intl
- ldap
- mcrypt (before PHP 7.2)
- mysqli
- opcache
- pdo_mysql
- recode (before PHP 7.4)
- snmp
- sockets
- tidy
- xmlrpc
- xsl
- zip

PHP (available) - oci8 - pdo_oci

PECL

- imagick

BUILD YOUR OWN IMAGE

3.1 Before you begin

Clone the repository

```
git clone https://github.com/rimelek/docker-php.git ./rimelek-docker-php
cd ./rimelek-docker-php
```

Using the repository the only thing you need to do is specify which extensions you want to be installed. You can also install every supported extension if you do not specify any extension to install.

The "common" directory is a collection of files that will be copied to each specific PHP version's directory when you run `./prepare.sh -s PATH`. Replace PATH with the actual path to a PHP version.

Example

```
./prepare.sh -s 8.1/fpm
```

Note: You can also pull the already built images from Docker Hub.

3.2 oci8 and pdo_oci extensions

If you need to use `oci8` and `pdo_oci` extensions, you have to download some files from Oracle's website. Follow the steps below to install the extensions

- Download the Basic and SDK package of Oracle Instant Client from the official source (you may need to accept a license): Instant client

- Copy the contents of `instantclient_xx.x` directory from each package into `common/opt/oracle/instantclient`

- You can build the image now using `prepare.sh` and Docker. See in the next chapter.

Note: Faster way is building a custom image from existing images on Docker Hub, although you could end up having more extensions than you really need.

```
FROM rimelek/php:8.1-fpm

COPY oracle/instantclient /opt/oracle/instantclient

RUN re-php-install-ext oci8 && re-php-install-ext pdo_oci
```

In this case copy the contents of `instantclient` directory to `./oracle/instantclient`

Then you can save the new image into your own private repository.

3.3 Build an image

Make sure the project root folder is writable as the the script needs to create a folder called "build"!

Install all supported extensions:

```
./prepare.sh -s 8.1/fpm
cd build/8.1/fpm
docker build -t myphpimage .
```

or install only some specified additional extensions:

```
./prepare.sh -s 8.1/fpm
cd build/8.1/fpm
docker build --build-arg PHP_EXT_GROUP=none PHP_EXT="gd bz2" --build-arg PECL_EXT_
↪GROUP=none -t myphpimage
```

You can change 8.1 to any supported version like 5.6, 7.0, 7.1, 7.2, 7.3, 7.4, 8.0