

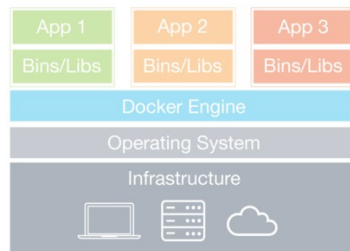
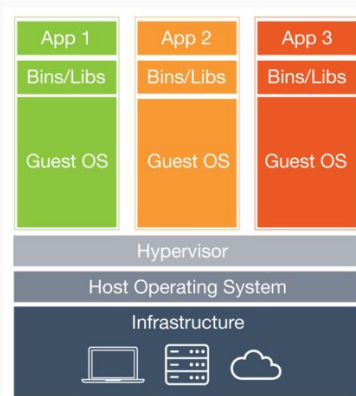
docker

DevCouch 05/05/2016

What is Docker?

Docker containers wrap up a piece of software in a complete filesystem that contains everything it needs to run: code, OS, php, node (etc) – anything you can install on a server.

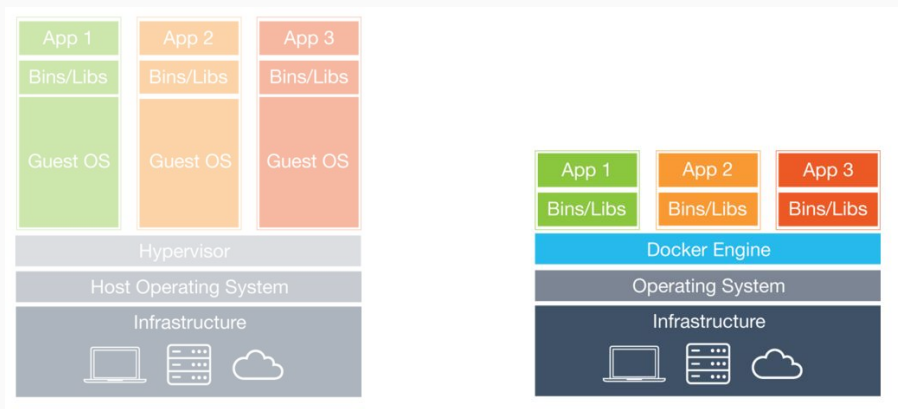
Vagrant vs Docker



Vagrant

- Downloads/Installs an OS and programmes for every machine you create
- Uses more disk space
- Slow build times

Vagrant vs Docker



Docker

- Share resources between containers (OS and libraries)
- Quicker build times
- Less disk space
- Ability to create versions and images

Docker Toolbox



Docker toolbox includes everything you need to get started

- Docker command line
- Docker engine
- Docker compose (?)
- And a program called kitematic

Apply for docker (beta) instead - <https://beta.docker.com/>

Dockerfile

```
FROM debian:latest
MAINTAINER Luke Barratt luke.barratt@creare.uk
```

```
# Add dotdeb.org package
RUN echo 'deb http://packages.dotdeb.org jessie all' >> /etc/apt/sources.list
RUN echo 'deb-src http://packages.dotdeb.org jessie all' >> /etc/apt/sources.list
```

```
ADD https://www.dotdeb.org/dotdeb.gpg .
RUN apt-key add dotdeb.gpg
```

```
RUN apt-get update -y
RUN apt-get install -y nginx php-fpm php-mysql php-curl curl git redis-server nodejs npm cron rsyslog supervisor
php7.0-gd vim nano
```

```
RUN echo "daemon off;" >> /etc/nginx/nginx.conf
RUN echo "cgi.fix_pathinfo=0" >> /etc/php/7.0/fpm/php.ini
RUN sed -i -e "s/^memory_limit\s=\s128M/memory_limit = 512M/" /etc/php/7.0/fpm/php.ini
RUN sed -i -e "s/^;clear_env\s=\sno/clear_env = no/" /etc/php/7.0/fpm/pool.d/www.conf
RUN sed -i -e "s/^;daemonize\s=\syes/daemonize = no/" /etc/php/7.0/fpm/php-fpm.conf
RUN sed -i -e "s/^bind\s*127.0.0.1/#bind 0.0.0.0/" /etc/redis/redis.conf
RUN sed -i -e "s/^variables_order\s*=\s*\\"GPCS\\"/variables_order=\\"EGPCS\\"/" /etc/php/7.0/fpm/php.ini
RUN sed -i -e "s/^variables_order\s*=\s*\\"GPCS\\"/variables_order=\\"EGPCS\\"/" /etc/php/7.0/cli/php.ini
ADD . /var/www/html
RUN ln -s /usr/bin/nodejs /usr/bin/node
RUN cd /var/www/html/public; npm install
```

```
ADD docker_files/default /etc/nginx/sites-available/default
ADD docker_files/envs.php /opt/envs.php
ADD docker_files/supervisor /etc/supervisor/conf.d
RUN touch /var/www/html/.env
```

```
RUN php -f /opt/envs.php
```

```
RUN service php7.0-fpm start
```

```
RUN curl -sS https://getcomposer.org/installer | php -- --install-dir=/usr/local/bin --filename=composer
```

```
RUN composer --working-dir=/var/www/html install --prefer-source --no-interaction
```

```
RUN chgrp -R www-data /var/www/html
```

```
RUN chmod -R 777 /var/www/html/storage
RUN chmod -R 777 /var/www/html/vendor/mpdf
RUN chmod -R 775 /var/www/html/bootstrap/cache
```

```
RUN crontab -l | { cat; echo " * * * * * php /var/www/html/artisan schedule:run"; } | crontab -
RUN cron -L15
```

```
EXPOSE 80 3000 3001 6379
```

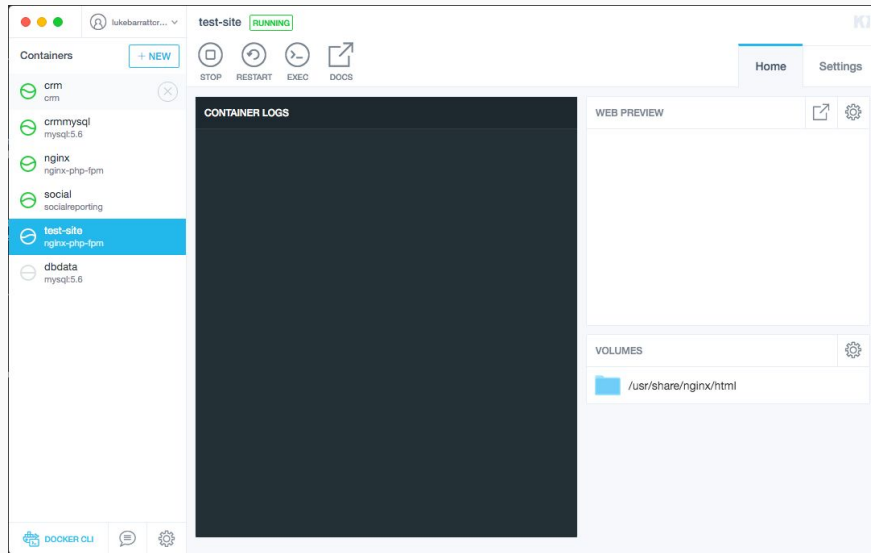
```
CMD (php -f /opt/envs.php && /usr/bin/supervisord -n)
```

Building and Running

- Navigate to folder with dockerfile
- `docker build -t lukebarrattcreare/crm .`
- `docker run -p 80:80 --name crm -v $(pwd):/var/www/html -d lukebarrattcreare/crm`

More simple way to get a container

- `docker pull richarvey/nginx-php-fpm`
(example of nginx and php container)
- `docker run -p 8080:80 --name test-site -v $(pwd):/usr/share/nginx/html -d richarvey/nginx-php-fpm`
- Your site should then be at your docker ip which is usually - `http://192.168.99.100:8082/`



Any questions?
Come ask me!