"¿Qué me estás container?"
Docker for dummies

Sara Arjona
@sara_arjona
Developer at Moodle HQ

Pau Ferrer
@crazyserver
Moodle Mobile developer at Moodle HQ

#MootES18
Entorno de desarrollo rápido con Vagrant

http://goo.gl/4HPWpz

Y el año que viene...

Entorno de desarrollo rápido con Docker
Questions about Docker

- Who knows about Docker?
- Who uses Docker for development?
- Who uses Docker in production?
- Who tried but could not do it?
- Who is interested in?
Docker

- Open platform for developing, shipping and running applications
- Allows package an application with all its dependencies into a standardized unit
- Provides abstraction of operating-system level virtualization
- Founded in 2013 as linux developer tool by Solomon Hykes (@solomonstre)

https://www.docker.com/
VM vs Container

- Containers are isolated but share OS and, when appropriate, bins/libraries
- There is no OS required inside of a container to run inside
Docker benefits

● Open Source
● Easy and fast deployment, migration, restarts
  ○ Microservices and integrations (APIs)
● Secure
● Lightweight (save disk, CPU & RAM)
  ○ No need to install guest OS
  ○ Run more apps on one host machine
● Great portability
  ○ Build in one environment, ship to another
● Simplify DevOps. Separation of concerns:
  ○ Developers focus on building apps
  ○ Sysadmins focus on deployment
● Version control capabilities
Docker workflow

- Pulls an image from the registry
- Creates a new container
- Allocates a filesystem and mounts a read-write layer
- Allocates a network/bridge interface
- Sets up an IP address
- Executes a process (if you specify it)
- Captures and provides application output
Who uses Docker?

Developers
- Local development
- Development collaboration
- Sandbox environment (develop, test, debug, educate)
- To eliminate the “works on my machine” problem

System operators
- Run and manage apps faster
- Continuous Integration & Deployment
- Scaling apps
- Infrastructure configuration

Enterprises
- Continuous Delivery Model
- Ship new features faster and more securely
Interesting Docker tool

- **Portainer**
  - Simple management UI for Docker
  - [http://portainer.io](http://portainer.io)
Docker & Moodle
Moodle Docker features

- All supported database servers (PostgreSQL, MySQL, Microsoft SQL Server, Oracle XE)
- All supported PHP versions (5.6, 7.0, 7.1)
- Behat/Selenium configuration for Firefox and Chrome
- Catch-all smtp server and web interface to messages using MailHog
- All PHP Extensions enabled configured for external services (e.g. solr, ldap)
- Zero-configuration approach
- Backed by automated tests

https://github.com/moodlehq/moodle-docker
Moodle Docker configuration

```bash
# Clone moodle-docker repository
git clone https://github.com/moodlehq/moodle-docker.git
cd moodle-docker

# Set up path to Moodle code
echo "export MOODLE_DOCKER_WWWROOT=/path/to/moodle/code"

# Choose a db server: pgsql, mariadb, mysql, mssql, oracle
export MOODLE_DOCKER_DB=mysql

# Ensure customized config.php for the Docker containers is in place
echo "cp config.docker-template.php $MOODLE_DOCKER_WWWROOT/config.php"
```

- Other environment variables:
  - MOODLE_DOCKER_PHP_VERSION (5.6, 7.0, 7.1)
  - MOODLE_DOCKER_BROWSER (for behat; default: firefox)
  - MOODLE_DOCKER_WEB_HOST (default: localhost)
  - MOODLE_DOCKER_WEB_PORT (default: 8000)
  - MOODLE_DOCKER_SELENIUM_VNC_PORT (e.g. 5900)
  - MOODLE_DOCKER_PHPUNIT_EXTERNAL_SERVICES (e.g. true)
# Start up containers

$ bin/moodle-docker-compose up -d

Creating network "moodledocker_default" with the default driver
Creating moodledocker_exttests_1 ...
Creating moodledocker_db_1 ...
Creating moodledocker_mailhog_1 ...
Creating moodledocker_selenium_1 ...
Creating moodledocker_mailhog_1
Creating moodledocker_exttests_1
Creating moodledocker_db_1
Creating moodledocker_selenium_1 ... done
Creating moodledocker_webserver_1 ...
Creating moodledocker_webserver_1 ... done

# Wait for DB to come up

$ bin/moodle-docker-wait-for-db
Moodle docker commands

- Stop containers
  
  ```
  $ bin/moodle-docker-compose stop
  ```

- Start containers

  ```
  $ bin/moodle-docker-compose start
  ```

- Shut down and destroy containers

  ```
  $ bin/moodle-docker-compose down
  ```

- Access into a container

  ```
  $ docker exec -it container_name bash
  ```
Moodle Docker for **manual testing**

- Initialize database

```bash
$ bin/moodle-docker-compose exec webserver php 
admin/cli/install_database.php --agree-license 
--fullname="Docker moodle" --shortname="docker_moodle" 
--adminpass="test" --adminemail="admin@example.com"
```

- Default URL: http://localhost:8000/
Moodle Docker for **phpunit** testing

- Initialize phpunit environment
  
  ```bash
  $ bin/moodle-docker-compose exec webserver php admin/tool/phpunit/cli/init.php
  [...]  
  ++ Success ++
  
  PHPUnit test environment setup complete.
  ```

- Run tests
  
  ```bash
  $ bin/moodle-docker-compose exec webserver vendor/bin/phpunit
  Moodle 3.6dev (Build: 20180614), 9e7c3978895c7cab585c2f5234ca536151d3bef6
  Php: 7.1.9, mysqli: 5.7.20, OS: Linux 4.13.0-45-generic x86_64
  PHPUnit 6.5.8 by Sebastian Bergmann and contributors.
  
  ..
  
  Time: 40.35 seconds, Memory: 36.00MB
  
  OK (2 tests, 7 assertions)
Moodle Docker for **behat** testing

- **Initialize behat environment**

  ```
  $ bin/moodle-docker-compose exec webserver php admin/tool/behat/cli/init.php
  [...]  
  ++ Success ++
  Acceptance tests site installed
  Acceptance tests environment enabled on http://webserver, to run the tests use:
  vendor/bin/behat --config /var/www/behatdata/behatrun/behat/behat.yml
  ```

- **Run tests**

  ```
  $ bin/moodle-docker-compose exec webserver php admin/tool/behat/cli/run.php
  --tags=auth_manual
  Running single behat site:
  Moodle 3.6dev (Build: 20180614), 9e7c3978895c7cab585c2f5234ca536151d3bef6
  Php: 7.1.9, mysqli: 5.7.20, OS: Linux 4.13.0-45-generic x86_64
  Server OS "Linux", Browser: "firefox"
  Started at 20-06-2018, 16:30
  ...............
  2 scenarios (2 passed)
  15 steps (15 passed)
  0m55.51s (43.28Mb)
  ```

- **The behat faildump directory is exposed at**

  http://localhost:8000/_/faildumps/
Moodle Docker for **behat** testing

- Using VNC to view behat tests
  a. Set MOODLE_DOCKER_SELENIUM_VNC_PORT (e.g. 5900)
  b. Download a [VNC client](#)
  c. Enter 0.0.0.0:5900 as the port in VNC Viewer (password ‘secret’)  
  d. Run behat tests
Moodle Docker for mailing

- Mailhog is listening on [http://localhost:8000/_/mail](http://localhost:8000/_/mail)
  a. First time, don’t forget to initialise ‘No-reply address’ setting
Moodle Docker for SOLR (search)

- MOODLE_DOCKER_PHPUNIT_EXTERNAL_SERVICES=true needed
- Create the ‘moodle’ index:

  ```
  $ docker exec -it --user=solr moodledocker_solr_1 bin/solr
  create_core -c moodle
  [...] 
  Creating new core 'moodle' using command:
  [...]
  ```

- Configure solr, set up search engine and index all the content:
Moodle Docker for LDAP

- MOODLEDOCKER_PHPUNIT_EXTERNAL_SERVICES=true needed
  - Used https://github.com/larrycai/docker-openldap image
- Check LDAP is working:
  - $ docker exec moodledocker_ldap_1 ldapsearch -x -H ldap://localhost -b ou=Users,dc=openstack,dc=org -LL

- Configure Authentication LDAP server:
  - Host URL: ldap://172.18.0.9*
  - Distinguished name: cn=admin,dc=openstack,dc=org
  - Password: password
  - Contexts: ou=users,dc=openstack,dc=org
  - User attribute: uid
  - Data mapping (First name): cn
  - Data mapping (Email address): mail
  - Data mapping (Description): description
  - Data mapping (Department): ou

- Users:
  - rjsmith / rJsmiH
  - Larrycai / LarryCai
And the next year...

:-P
#MootES18

Sara Arjona
@sara_arjona
Developer at Moodle HQ

Pau Ferrer
@crazyserver
Moodle Mobile developer at Moodle HQ