Webfactory: Drupal +Docker

Sean Boran 4th June 2015

https://github.com/Boran/docker-drupal
sean@boran.com

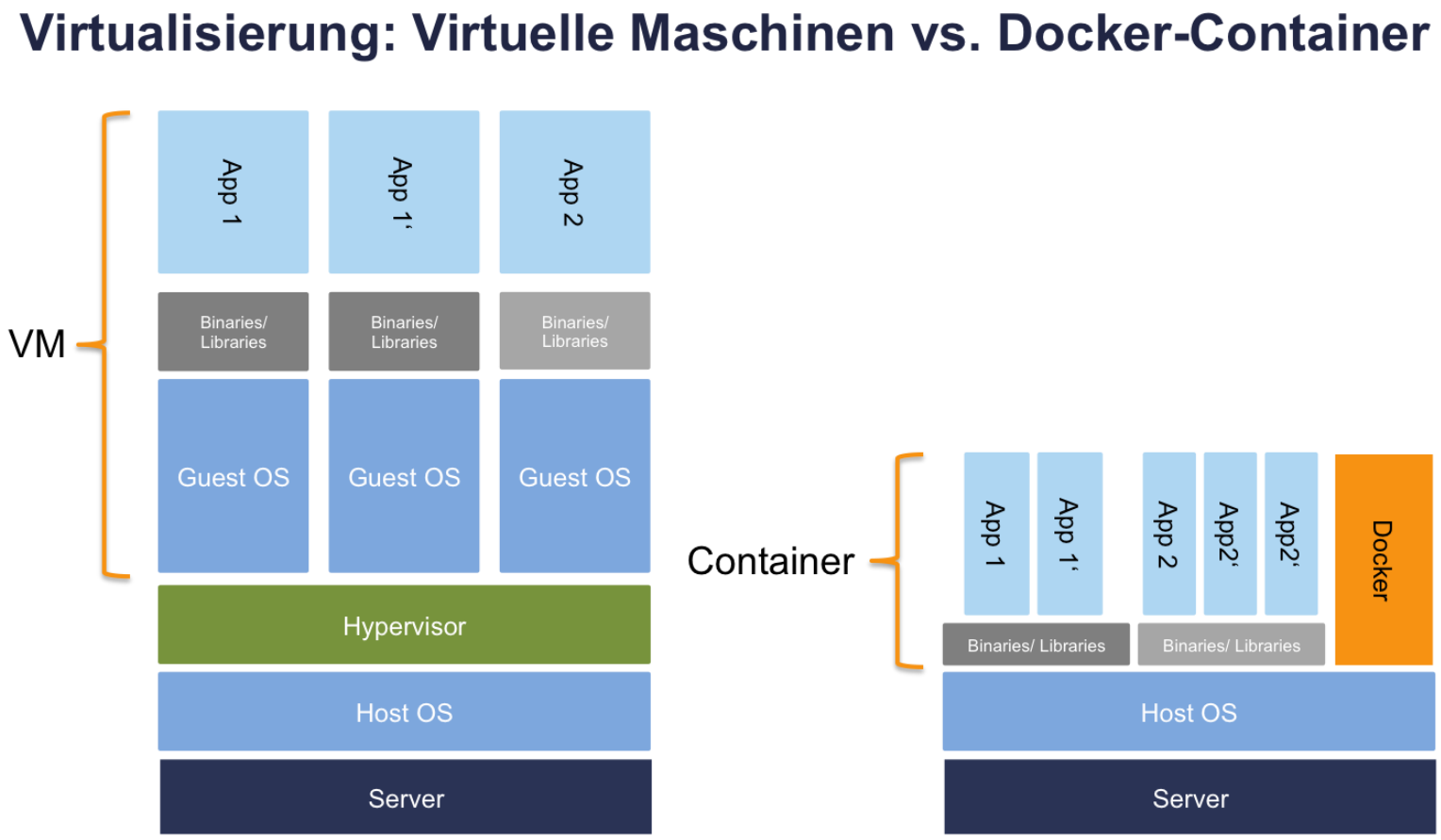
Presentation overview

- Aims
- Components & concepts
- Webfactory UI
- Docker containers & scripts
- Demo (if time)

Aims

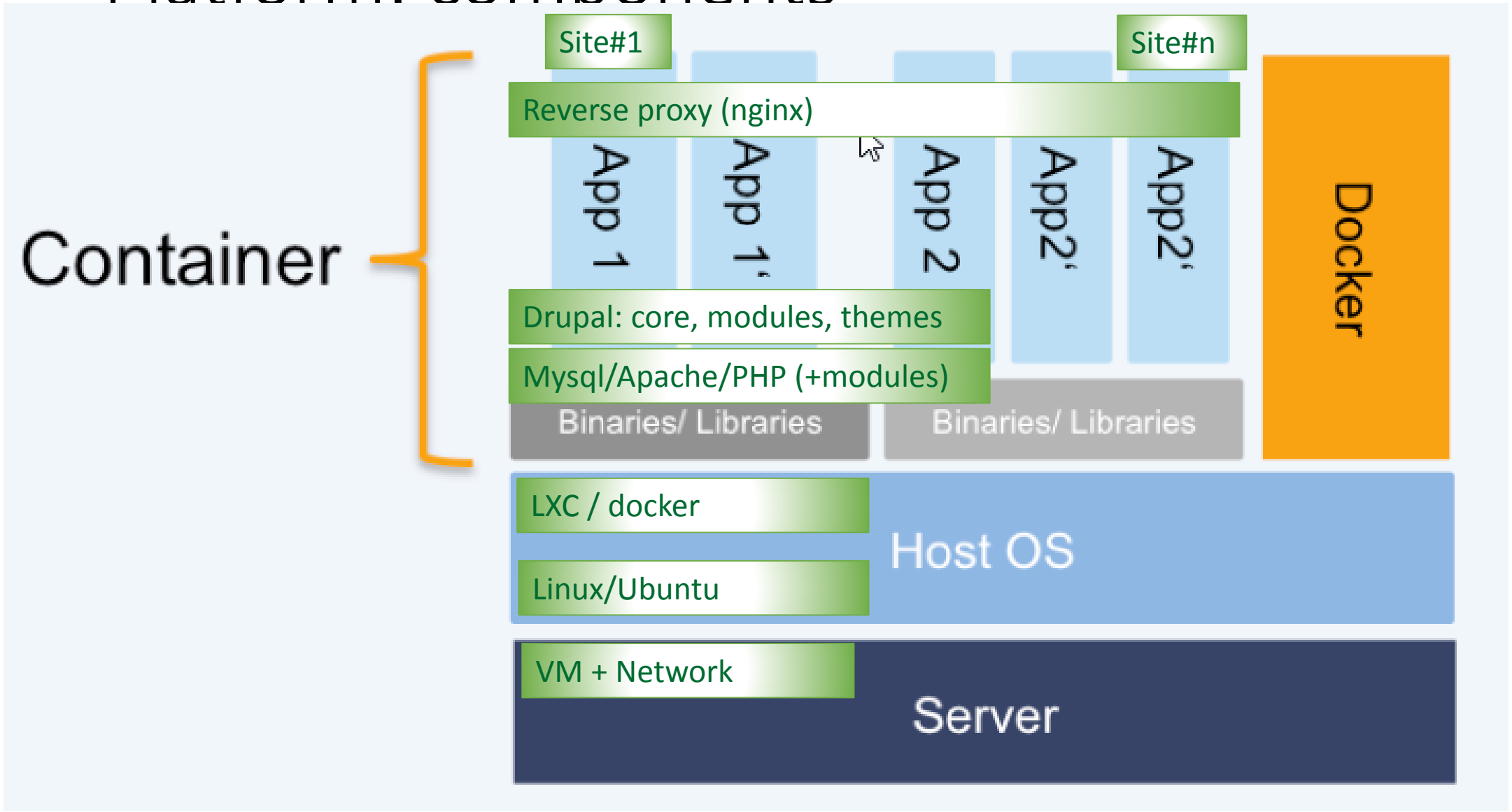
- Create a new website … in minutes .. without a techie!
- Pre-determined look/feel
- Easy to use
- Lean: low cost
- Hosted on Internal infrastructure

Platform: Use Virtualisation + docker containers

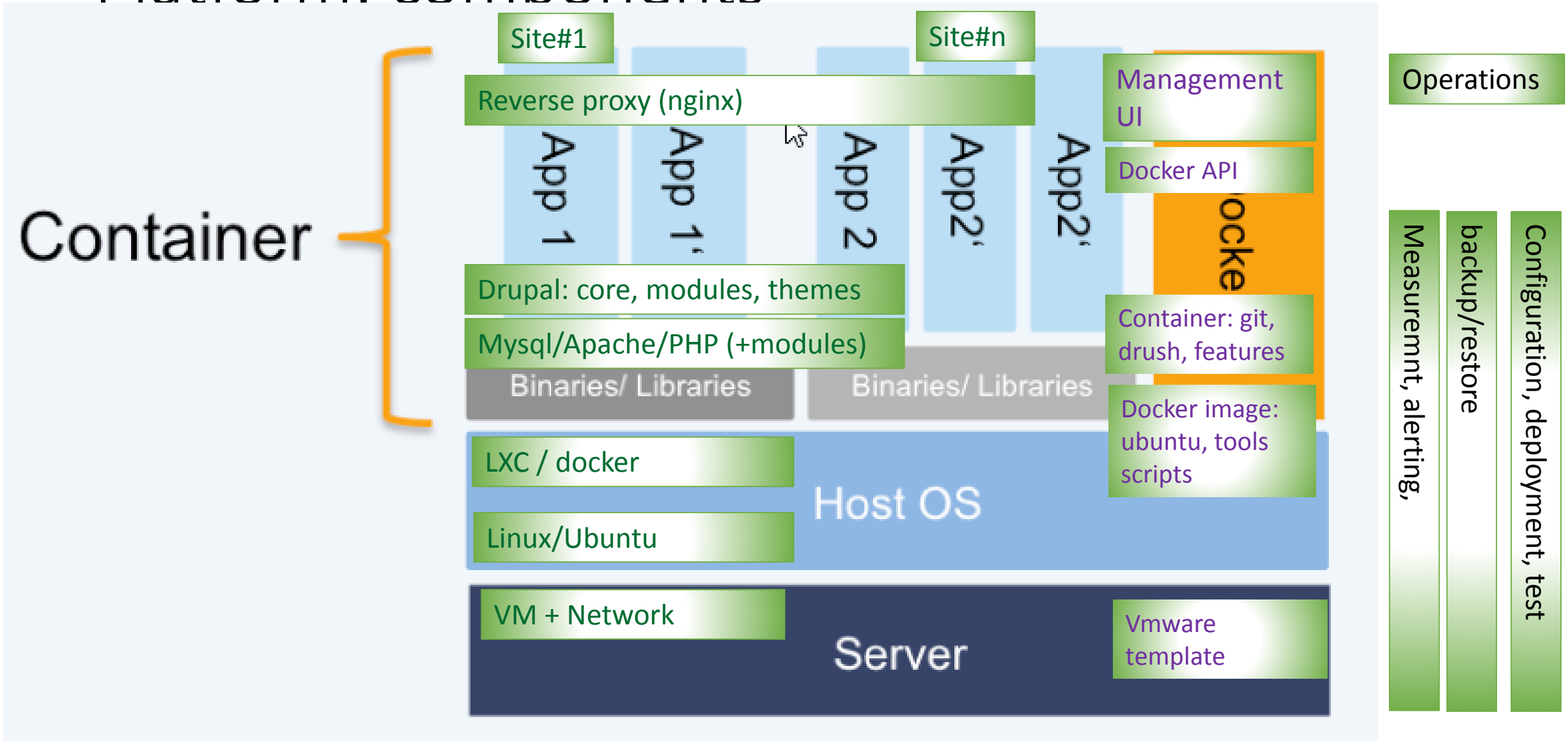


Quelle: Docker, Crisp Research, 2014

Platform: components



Platform: components



Webfactory: Admin UI

Webfactory test | Home | Websites | Templates | Contact | My account | Log out

Websites

	Website	Status	Menu	Owner	Category
! merge	merg.webfact.vptt.ch	stopped	Action	admin	Test
!novaauth	nova-auth.webfact.vptt.ch	stopped	Action	kkalicinski	Test
!Novahub	novahub.webfact.vptt.ch	stopped	Action	jonas	Test
!Starterkit Demo	starterkitdemo.webfact.vptt.ch	running	Action	jonas	Test
!starterkit27052015	starterkit27052015.webfact.vptt.ch	stopped	Action	jonas	Test
CIbuild Build server for nova/starterkit,...	cibuild.webfact.vptt.ch	running	Action	admin	Test
Incubation develop	incubationdev.webfact.vptt.ch	running	Action	jonas	Test
jenkins continuous building of Nova sites, port...	jenkins.webfact.vptt.ch	running	Action	admin	Test

UI: Add a website

Create Website

Title *
My hot site

Hostname *
mysite
The website will have this name prefixed to the webfactory domain (e.g. http://something.we
with numbers or a dash.

Site email *
boran@example.ch
Initial email for the website configuration ("from" address) and the "admin" user.

Template *
Plain drupal7
Template for building the website. Alternatively see the Advanced section below.

Category
Test
The Production categories indicates high availability requirements.

Auto-start
on-failure
Should the container be automatically started? The behaviour to apply when the container exi
container exit code is non-zero). Production containers should have "always" and test contair
changes to this setting.

ADVANCED

Custom docker configuration: If not using a template or to override template settings.

Docker Image
The docker base image to build the container from – a local or hub.docker.com name e.g. boran/drupal, i

DOCKER ENVIRONMENT
List of environment key=value pairs. Note: prefer setting environment variables on the template level.
Add another item

DOCKER VOLUMES
List of volumes and mount points e.g. /serverdir/ssh/id_rsa:/contdir/id_rsa:ro /serverdir/ssh/id_rsa.pub:
template level.
Add another item

DOCKER PORTS
Mapping of container exposed ports to public ports. format CONTAINER_PORT:SERVER_PORT, e.g. 80:800
Add another item

UI: Website status page

UI-Dev Home Websites Templates Contact My account Log out

Status Manage Advanced Backup/restore Docker Admin

Vanilla2

Meta data

Website:	vanilla2.webfact.vptt.ch
Category:	Test
Auto start:	on-failure
Owner:	admin
Description:	simple tests for sean

Actual

Run status:	running
Error:	(none)
Auto start:	(on-failure)

Run time:

App status:	Drupal7.35

Build status:	completed(100)

How

- Docker image «Boran-drupal»
  - Apache, php, (mysql)
  - Drupal installation/provision
  - Drupal post install
  - Site specific postfix, puppet, apache
- And
  - Nginx, nginx-gen
  - drupal-ci, jenkins
  - Inno-drupal (private)
- Webfactory UI:
  - Drupal, guzzle, bootstrap, JS, php.
- APIs
  - Docker API (go)
  - Docker-php

How #2

- https://github.com/Boran/
- DNS wildcard

- The Webfactory consists of several modules:
  - webfact-make (build/install)
  - webfact (main logic)
  - webfact_content_types (features/views)
  - webfact_theme (styling)

How is a docker container built?

- Create a container base on an image
- Optionally specify environment options

Create Drupal sites on the command line

- Create a D7 site
  docker run -td boran/drupal
- Drupal8 and map a port:

  docker run -td -p 8004:80 -e "DRUPAL_VERSION=drupal-8.0.0-beta10" --name d8 boran/drupal

  docker logs d8

- Read (much) more on https://github.com/Boran/docker-drupal

Install the webfactory: https://github.com/Boran/webfact-make

```
# install linux+docker. Then: pull image, build webfact container
docker pull boran/drupal
name=webfact
domain=$name.example.ch
email=bob@example.ch
image="boran/drupal"
port=8020
docker stop $name; docker rm $name;
docker run -td -p $port:80 -e "VIRTUAL_HOST=$domain" -v /var/run/docker.sock:/var/run/docker.sock -v
/opt/sites/$name:/data -v /opt/sites:/opt/sites -e "DRUPAL_SITE_NAME=WebFactory" -e
"DRUPAL_ADMIN_EMAIL=$email" -e "DRUPAL_SITE_EMAIL=$email" -e
"DRUPAL_MAKE_REPO=https://github.com/Boran/webfact-make" -e "DRUPAL_MAKE_DIR=webfact-
make" -e DRUPAL_INSTALL_PROFILE=webfactp -e DRUPAL_FINAL_SCRIPT=/opt/drush-
make/webfact-make/scripts/final.sh  -e "VIRTUAL_HOST=$domain" --restart=always --hostname
$domain --name $name $image
docker logs -f $name
```

How is the «boran/drupal» image built?

https://github.com/Boran/docker-drupal (open this URL yourself while we talk)

- Dockerfile: what OS+packages+files to install
  - Lamp
  - supervisor
- Start.sh: automate startup and provisioning
  - Install drupal
  - Mysql (optional)
  - supervisor
- example_run.txt
- example_api.txt

Nginx reverse proxy for port mapping

- Make containers available as a dynamic subdomain http://SOMETHING.mydomain.ch
- Dns wildcard
- Maps to port 80 on the appropriate container via the VIRTUAL_HOST and VIRTUAL_PORT environment.

```
docker run -d -p 80:80 -v /var/run/docker.sock:/tmp/docker.sock --restart=always --hostname nginxproxy --name nginxproxy jwilder/nginx-proxy
```

https://github.com/jwilder/docker-gen

Demo

Demo: Webfact #1

Logon to UI

1. Get «site owner» priviliges
2. Create a website:
    1. meta data
    2. Create (% complete)
3. Logs, inspect
4. Stop, start

5. Run command
6. Folder download

7. Delete
8. Delete all
9. Backup list, backup, restore, remove all

Get «admin» privileges

1. Nginx: logs, restart

2. Edit website template
3. Website update
4. Rebuild from sources
5. Rebuild from meta-data
6. Run container OS update

7. List containers, images

Demo Webfact #2

Get «admin» privileges

1. Edit website template
2. Website update
3. Rebuild from sources
4. **Rebuild from meta-data**
5. **Run container OS update**

6. List containers, images

7. Delete all, remove all

Inno/drupal

- What is does
- Puppet config
- Custom puppet config

Nginx: logs, restart, troubleshooting

Using other images:

- Jenkins
- Wordpress
- ??

Demo: Webfact settings

- Settings hierarchy:
  - Website
  - Template
  - server level.
- Webfact settings: admin/config/development/webfact
- Some «meta data» fields:
  - Template
  - Category: production/test
  - Auto-start
  - «advanced»
- DNS wildcard

Thank you

- Feedback ?
- Questions ?