Perl and Docker

Jon Allen (JJ) – jj@opusvl.com

OPUSVL

Docker

Packages an application and dependencies into a portable container

Concepts

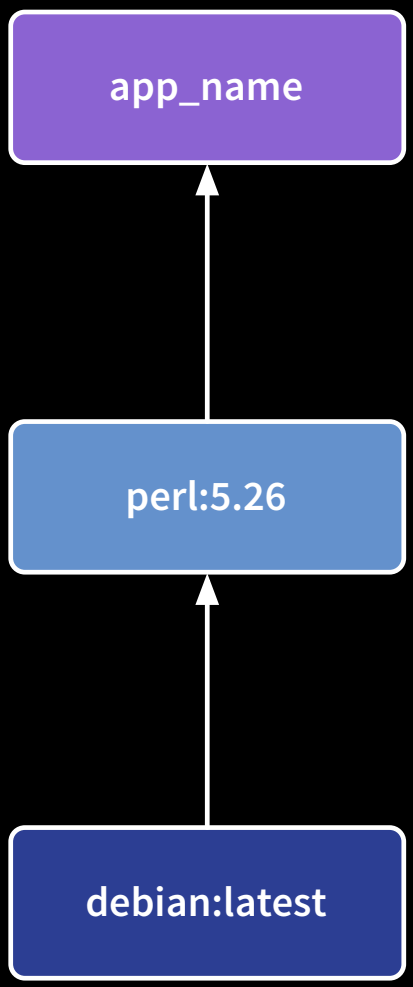
Dockerfile

Set of instructions to build an image

```
FROM quay.io/opusvl/fb11
MAINTAINER Alastair McGowan-Douglas <alastair.mcgowan@opusvl.com>

USER root

RUN apt-get update && apt-get -y install libsodium-dev && apt-get clean

RUN /opt/perl5/bin/cpanm -M http://cpan.opusvl.com/ -n Term::ReadKey
RUN /opt/perl5/bin/cpanm -M http://cpan.opusvl.com/ IO::Socket::SSL
RUN /opt/perl5/bin/cpanm -M http://cpan.opusvl.com/ -n OpusVL::CMS

RUN mkdir /opt/cms \
   && groupadd -r cms \
   && useradd --home /opt/cms -r -g cms cms \
   && chown -R cms: /opt/cms

ENV PERL5LIB=/opt/cms/lib/perl5:$PERL5LIB

USER cms
```

Layers

Inheritance and image re-use

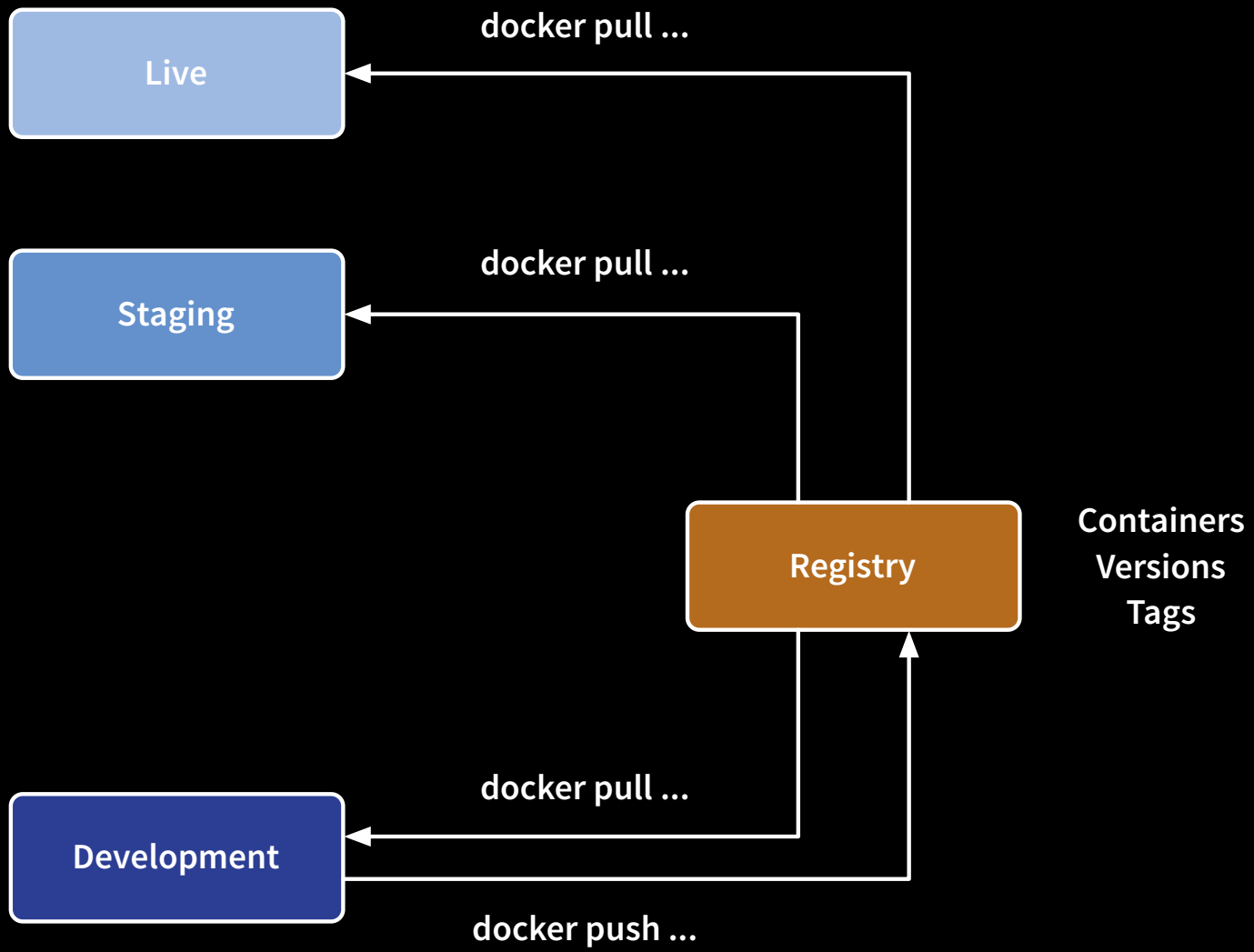



```
FROM perl:5.26
...
# install application
```

```
FROM debian:latest
...
# install Perl
```

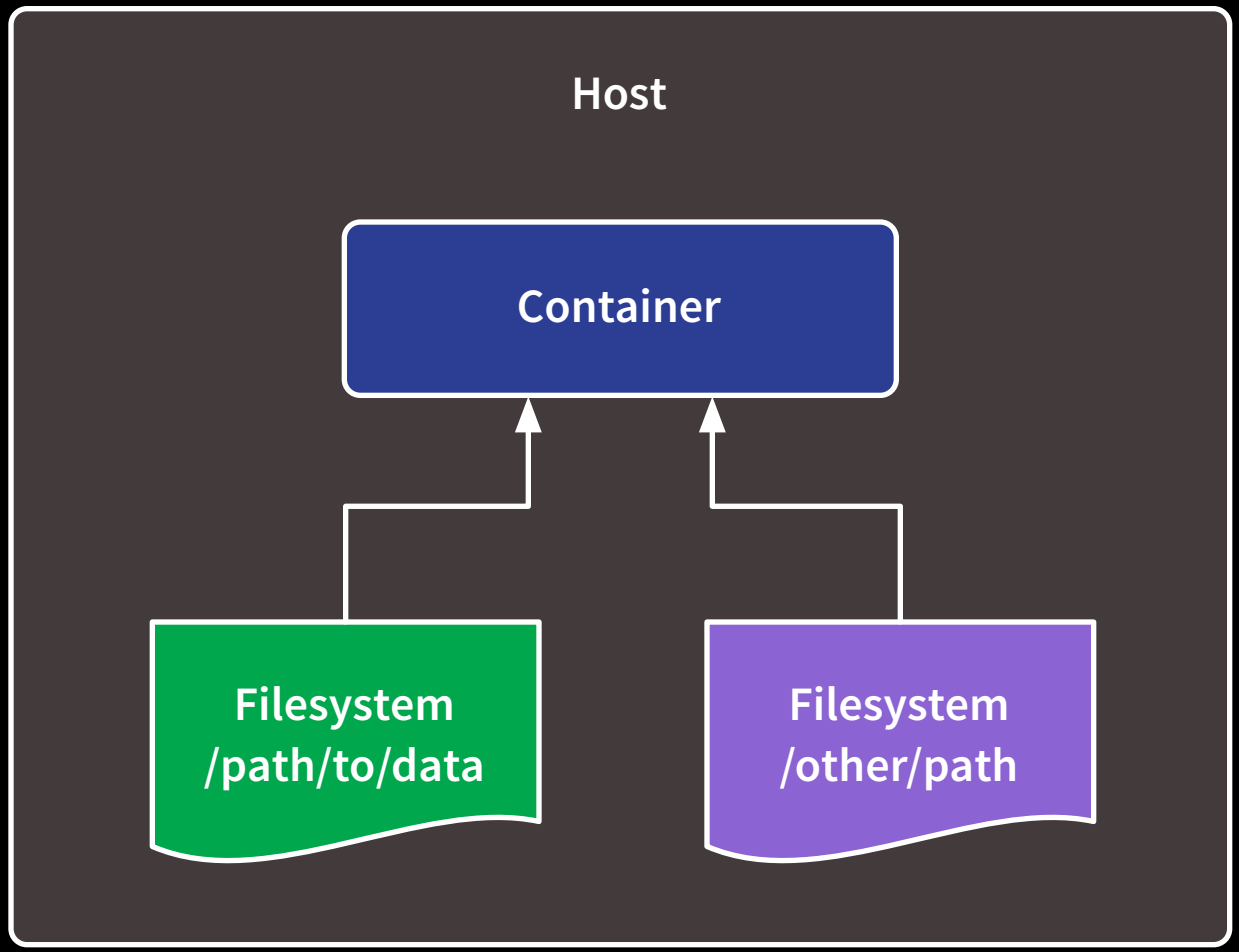
Registry

Version controlled repository of layers and images


Live
docker pull ...
Staging
docker pull ...
Registry
Containers
Versions
Tags
Development
docker pull ...
docker push ...

Volumes

Store persistent data outside of the container

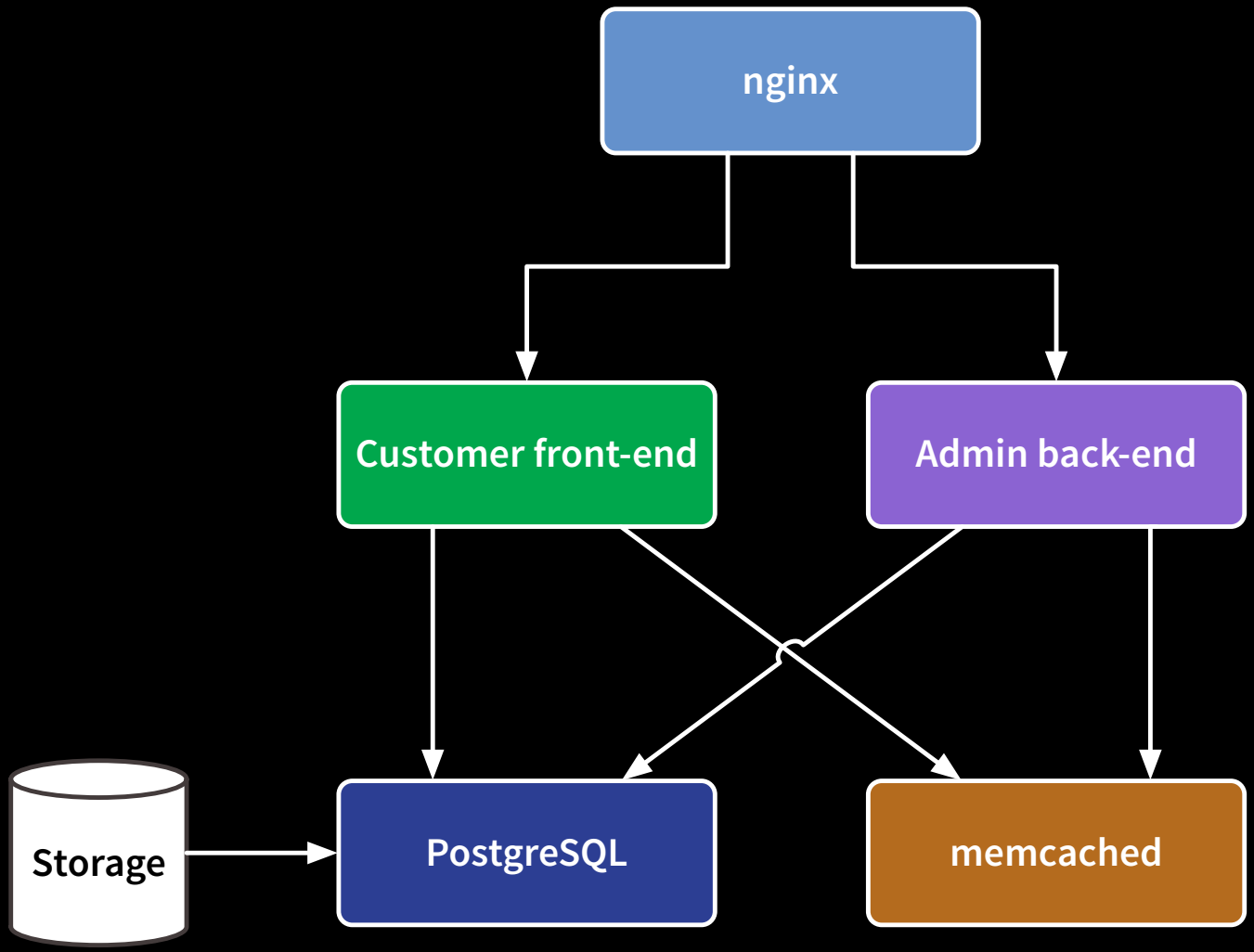

Host
Container
Filesystem
/path/to/data
Filesystem
/other/path

Entrypoint

Command that is run when the container starts

Docker Compose

Because applications have moving parts


nginx
Customer front-end
Admin back-end
Storage
PostgreSQL
memcached

```
version: '2'
services:
 db:
  image: quay.io/opusvl/postgres:9.4
  environment:
   PGDATA: /var/lib/postgresql/data/pgdata
   POSTGRES_USER: username
   POSTGRES_PASSWORD: "${POSTGRES_PASSWORD}"

 memcache:
  image: memcached

 website:
  image: quay.io/opusvl/appname_front_end
  environment:
   APPNAME_View__Email: '{"sender":{"mailer_args":{ ... }"}}}'

...
```

Configuration

Base configuration in docker-compose.yml, which is merged with override file

Convention is to use environment variables

Catalyst::Plugin::Config Loader::Environment

Why use Docker?

Fast

Portable

Standard

Docker and Perl

Three things we wanted to fix

1. Deployment

Live environments managed by Infrastructure team

No developer access

2. “Works on my machine”


WORKED FINE IN DEV
OPS PROBLEM NOW
memegenerator.net

“Works on my machine”

“So of course it will work in production”

3. Developer onboarding

Lots of projects, some shared dependencies, takes time to set up a new dev environment

Solution

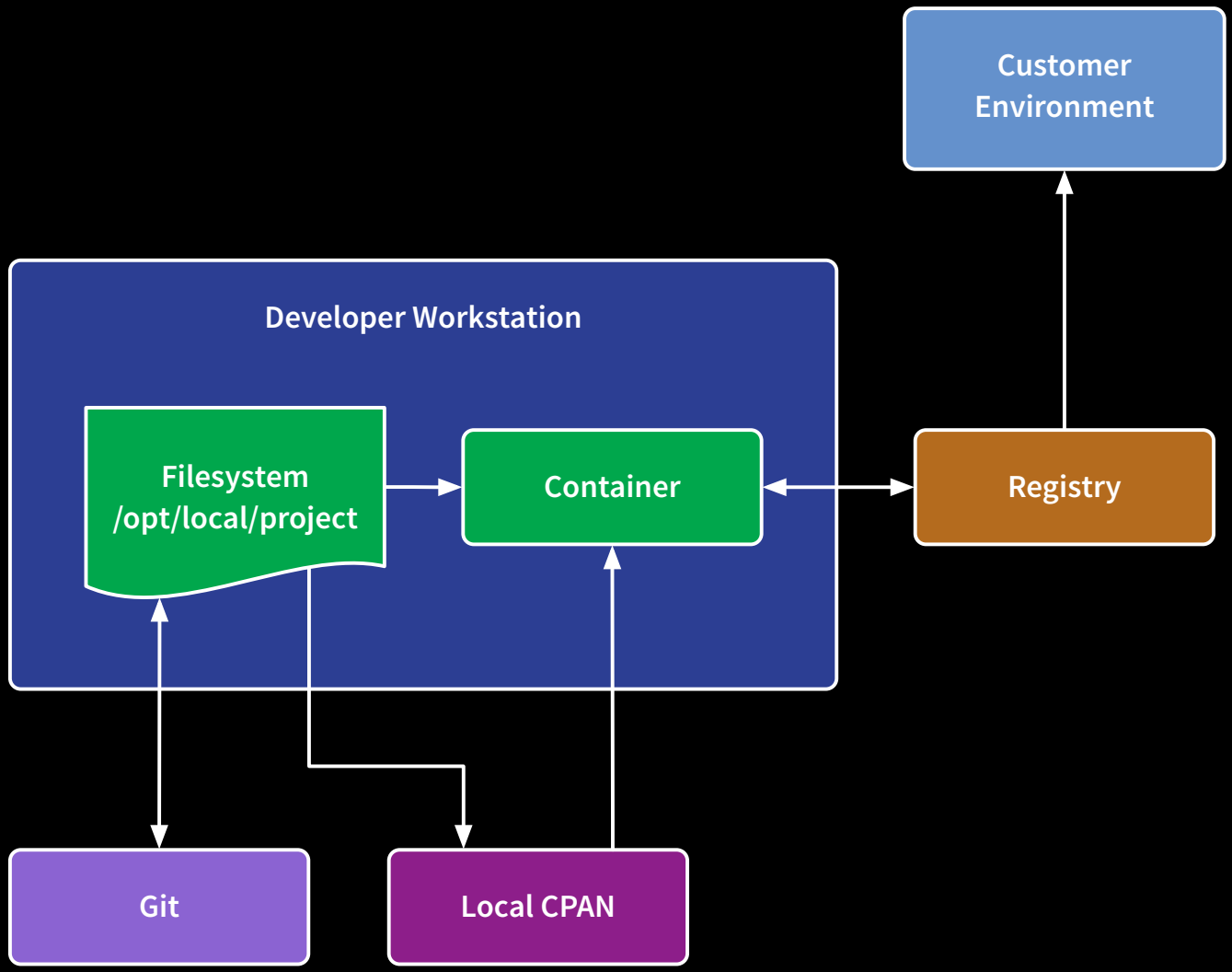
With Docker, we can run the same containers in Development, Staging, and Live

Same code

Same environment (from Docker Compose)

Developer workflow

Mount local Git repos and inject into container, overriding the installed code


Customer Environment
Developer Workstation
Filesystem /opt/local/project
Container
Registry
Git
Local CPAN

entrypoint.pl

If $ENV{DEV_MODE} is set…

Searches for volumes

/opt/local/project/dist/lib
/opt/local/*/*/lib

Adds each lib/ directory found to $ENV{PERL5LIB}

Check out all repos to /opt/local/projectname/*

Mount volumes in docker-compose.override.yml

Automatically installs dependencies if $ENV{INSTALLDEPS} set

Uses plackup instead of Starman / Martian (single worker process)

github.com/ OpusVL/Perl-Docker