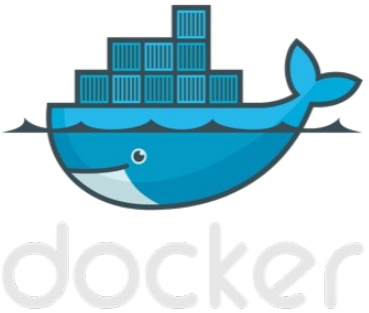
OFBiz Development with Docker



http://ofbiz.apache.org

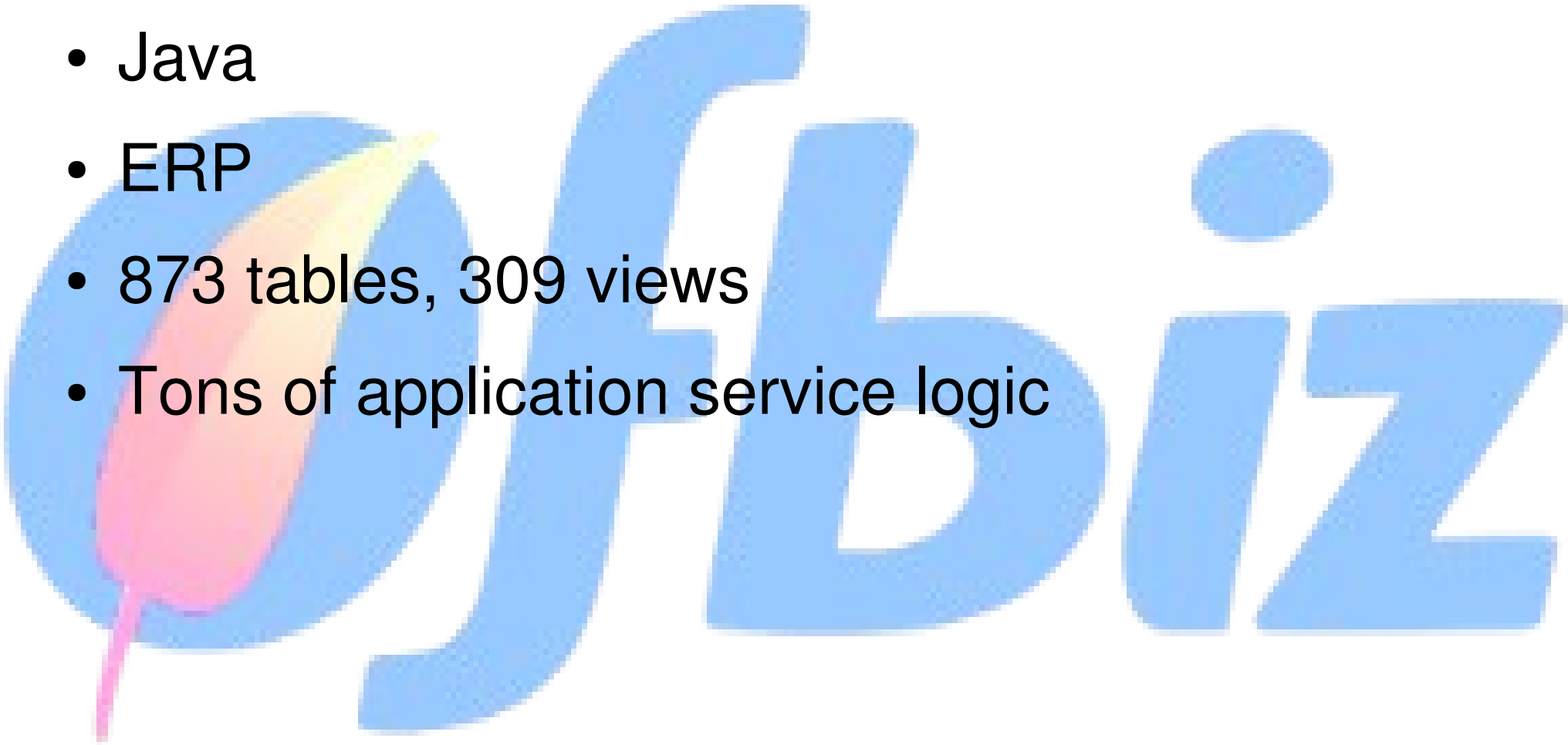
http://docker.io/

What is OFBiz?

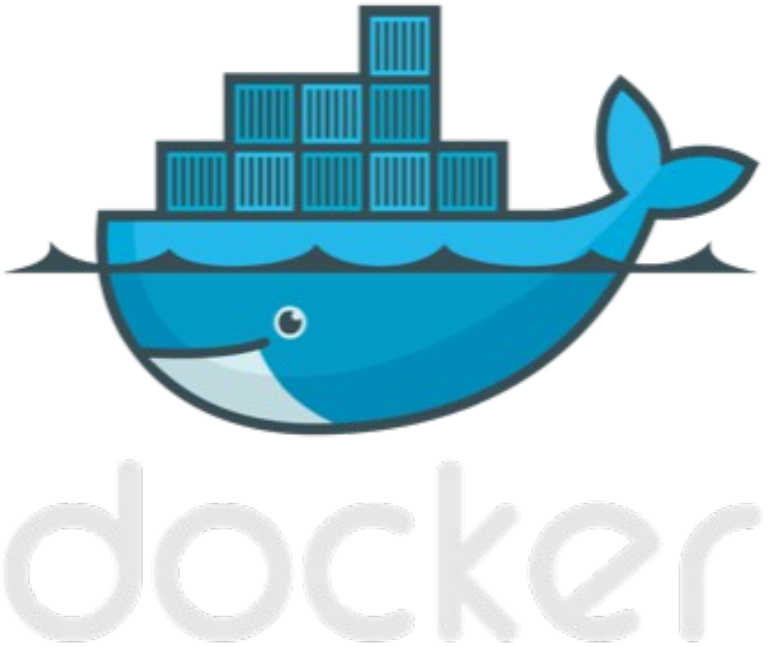
http://ofbiz.apache.org/

OFBiz

- Java
- ERP
- 873 tables, 309 views
- Tons of application service logic



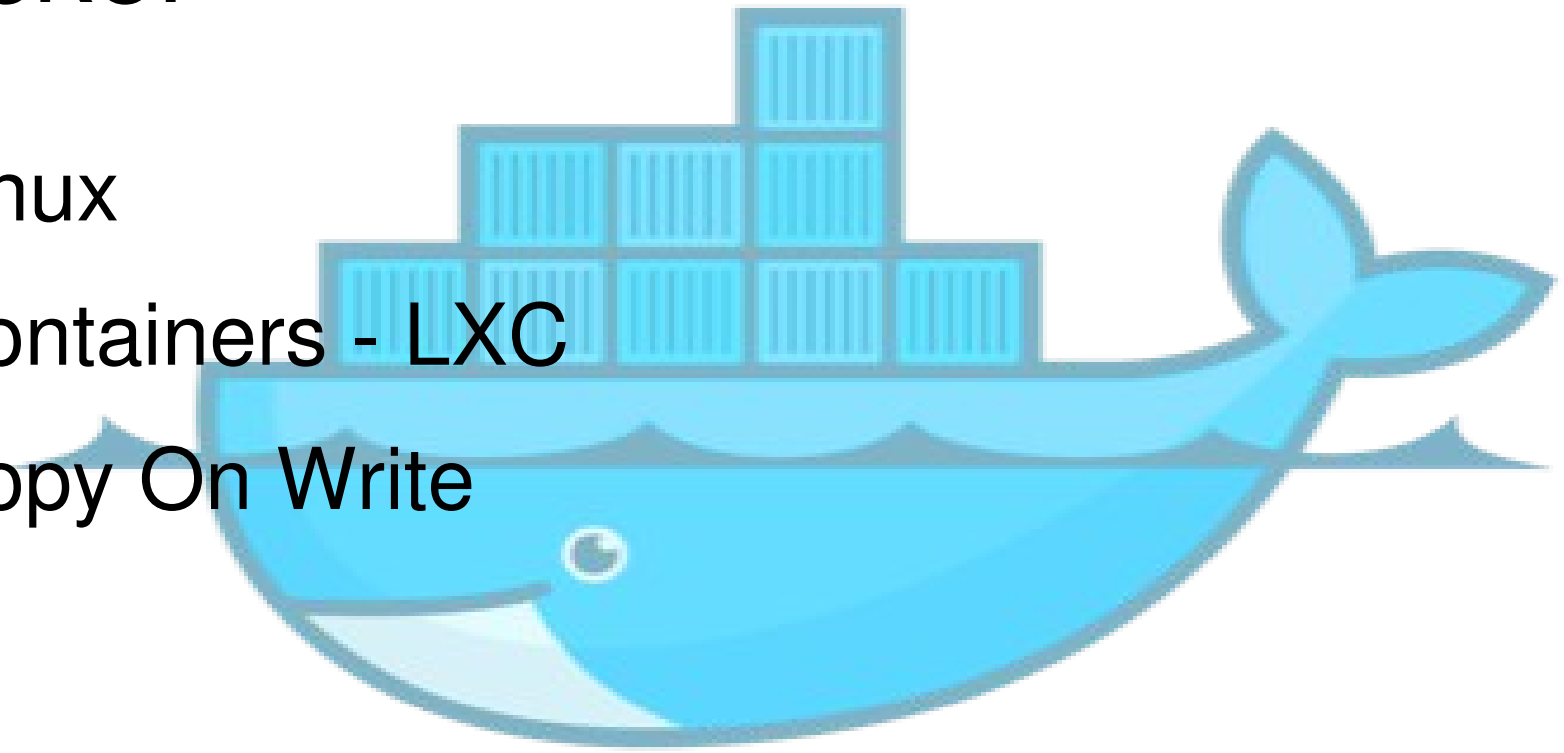
What is docker?



http://docker.io/

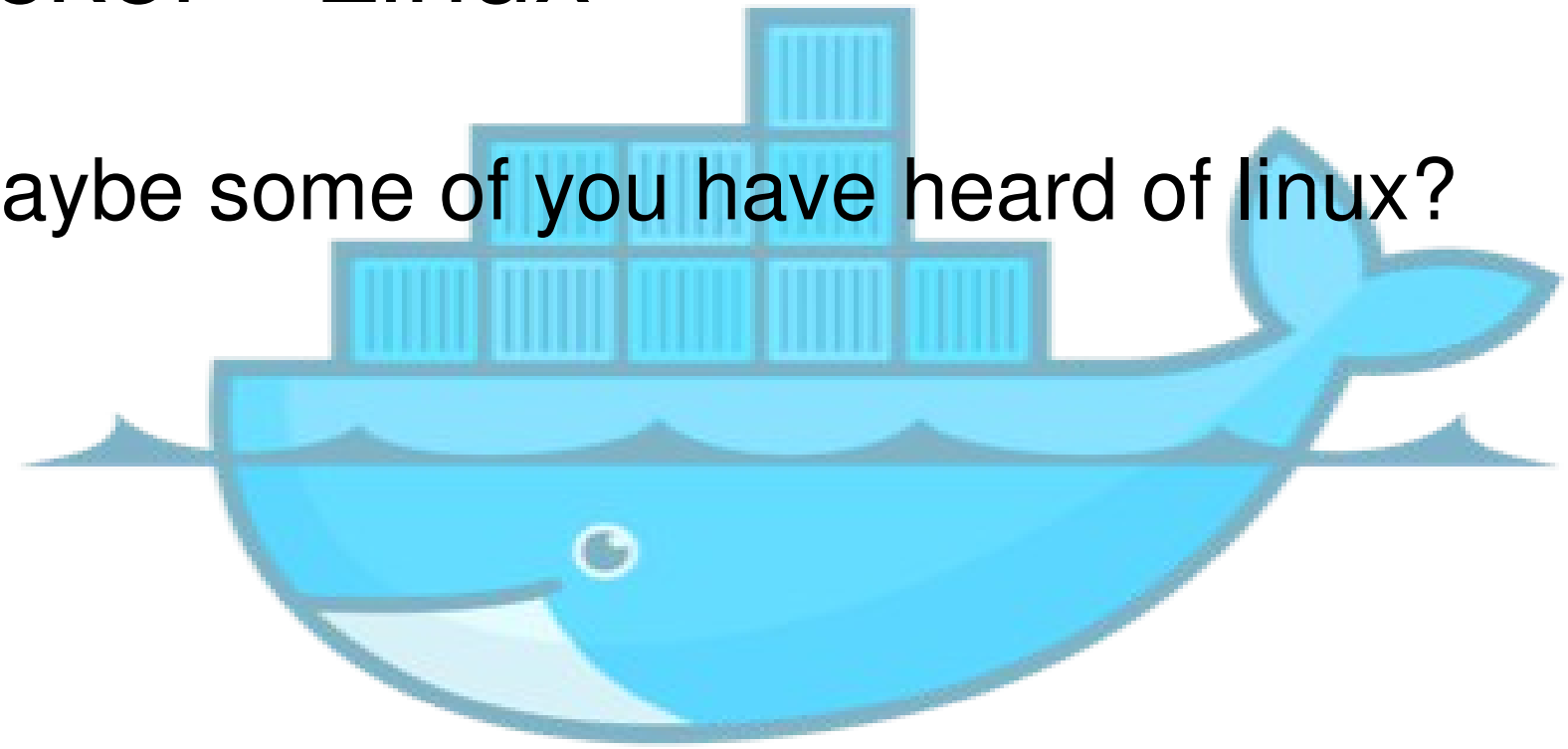
Docker

- Linux
- Containers - LXC
- Copy On Write



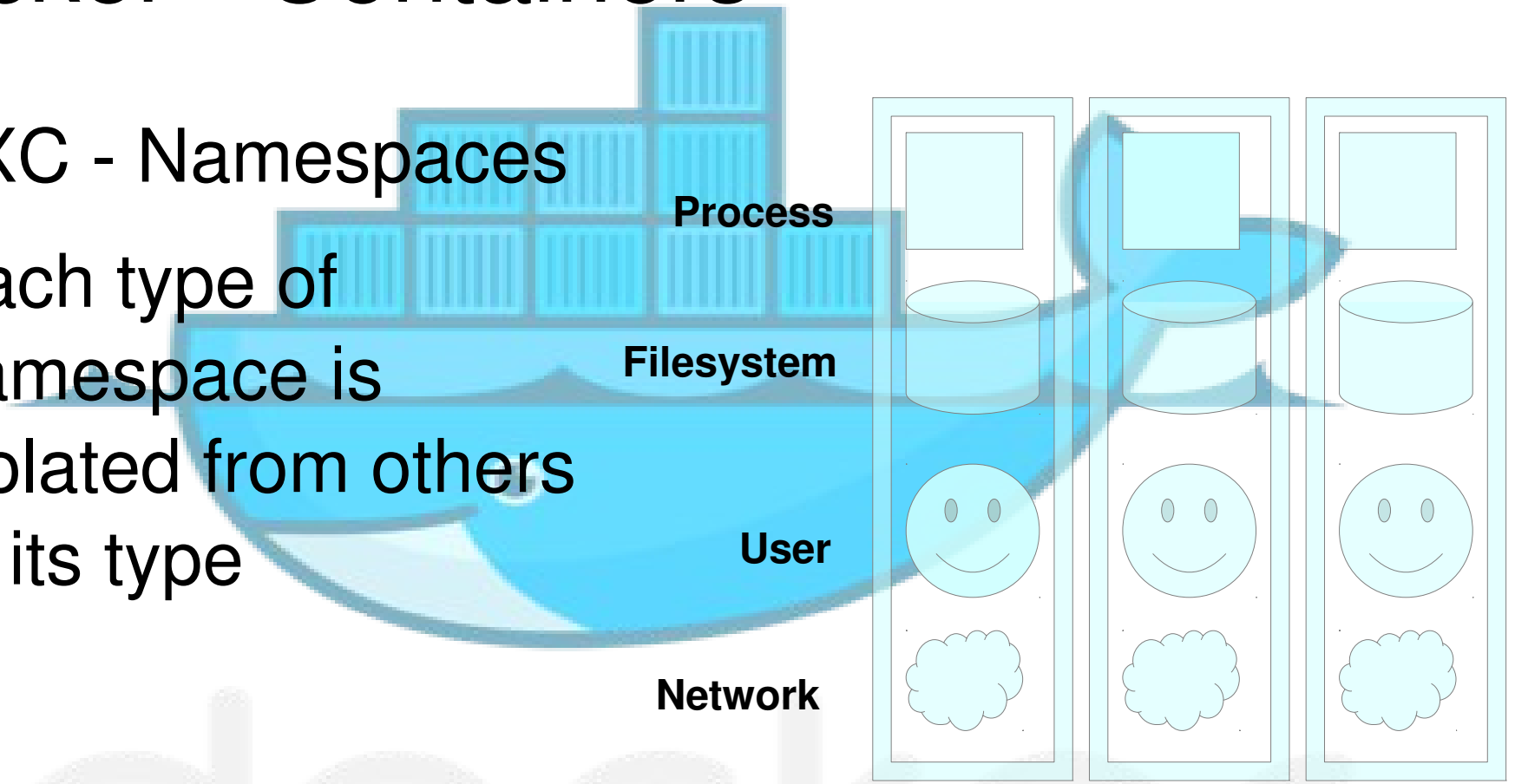
Docker - Linux

- Maybe some of you have heard of linux?



Docker - Containers

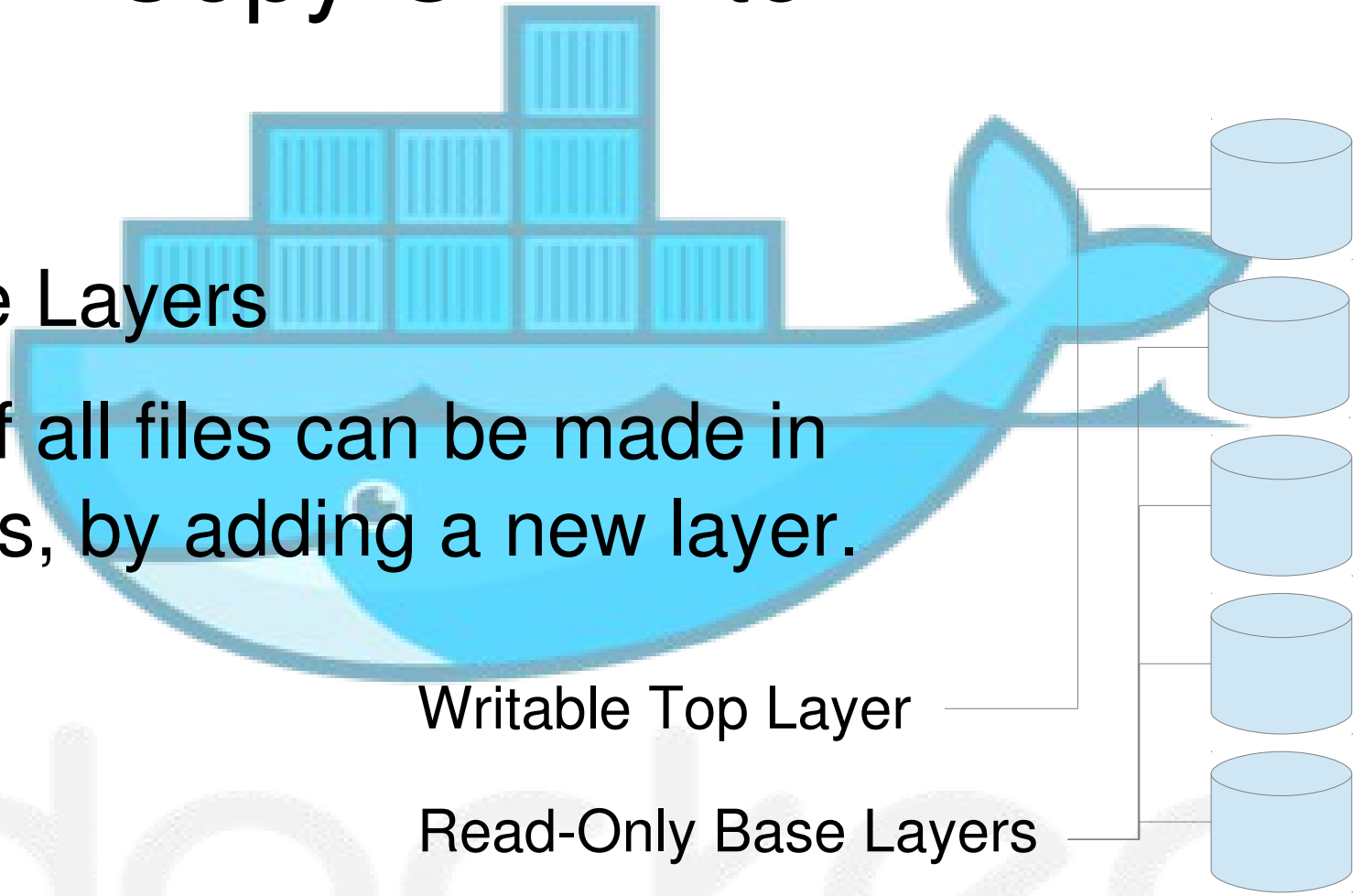
- LXC - Namespaces
- Each type of namespace is isolated from others of its type



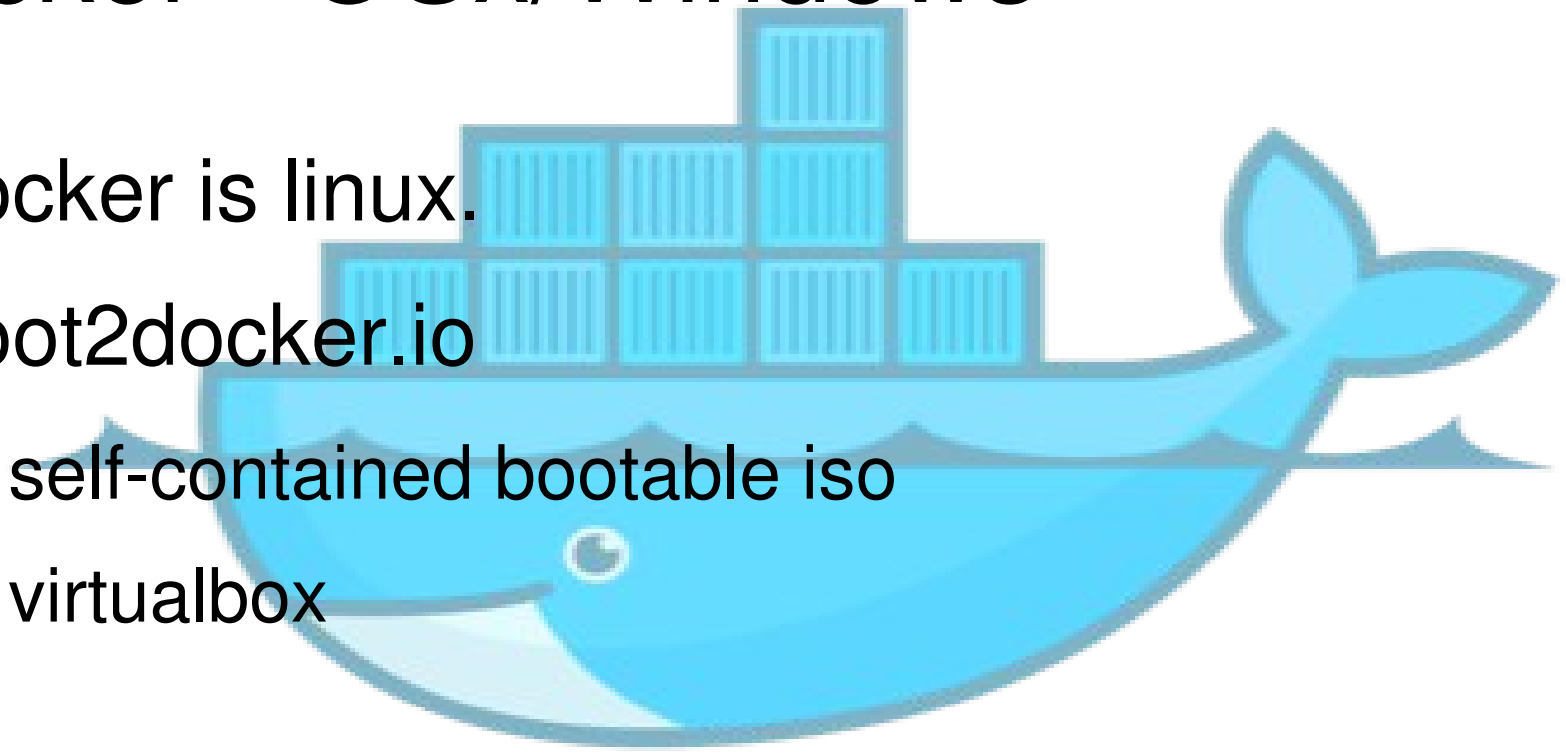
Lighter weight than standard virtualization - Better performance

Docker - Copy On Write

- AUFS
- Multiple Layers
- Copy of all files can be made in seconds, by adding a new layer.



Docker - OSx/Windows



- docker is linux.
- boot2docker.io
  - self-contained bootable iso
  - virtualbox

OFBiz Development - New Feature

- Get the code
- Initialize the system(*)
- Write the test case
- Implement the feature
- Submit for approval(and merge)

OFBiz Development - Bug Found

- Yes, bugs can happen.
- Replicate environment(*)
- Update test case
- Fix the code
- Submit for approval(and merge)
- Cross fingers, hope it works in production

OFBiz Development - Coding

- Which code base is being updated?
  - OFBiz backend, content frontend, high-availability cat herding dispatcher?
- Does the new code require changes in multiple layers?
- Which test case framework will get updated?
  - OFBiz, selenium, spreadsheet?

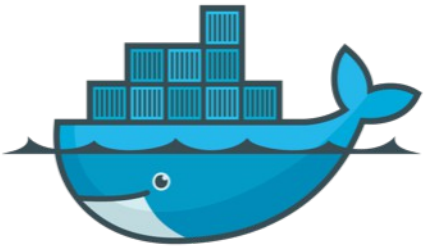
OFBiz Development - Environment

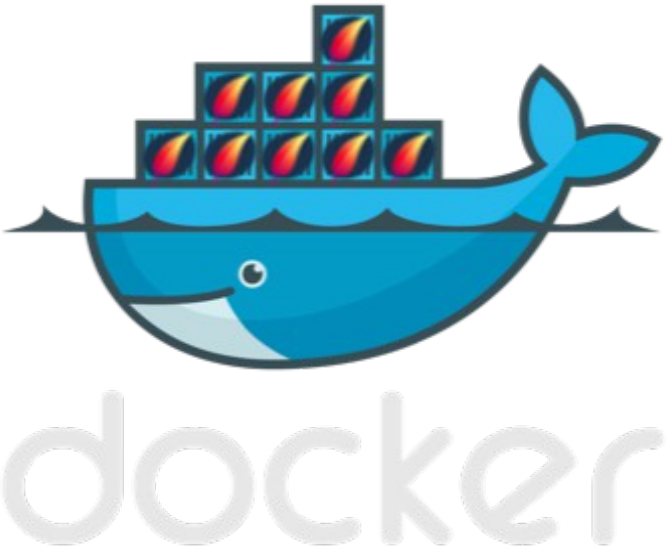
- Initialize and replicate require a real environment
- Which database?
  - Mysql, Postgresql, Derby, Oracle
- What web frontend?
  - Apache, Nginx
- Email(MTA)?
  - Exim, Postfix, Qmail, Sendmail

OFBiz Development - Environment

- Is there a content frontend?
  - wordpress, drupal, django, etc?
- How much data should be made available?
  - Empty, seed, or a full database copy?

OFBiz Development with Docker





docker - Dockerfile

- A simple way to run a series of commands
- Can do almost anything
  - Install packages
  - Modify files
  - Copy external configuration settings

docker - Dockerfile

- Each step is cached
  - This allows for long running steps to be skipped if nothing has changed
- There are many pre-packaged docker images available, to save time from having to build your own.

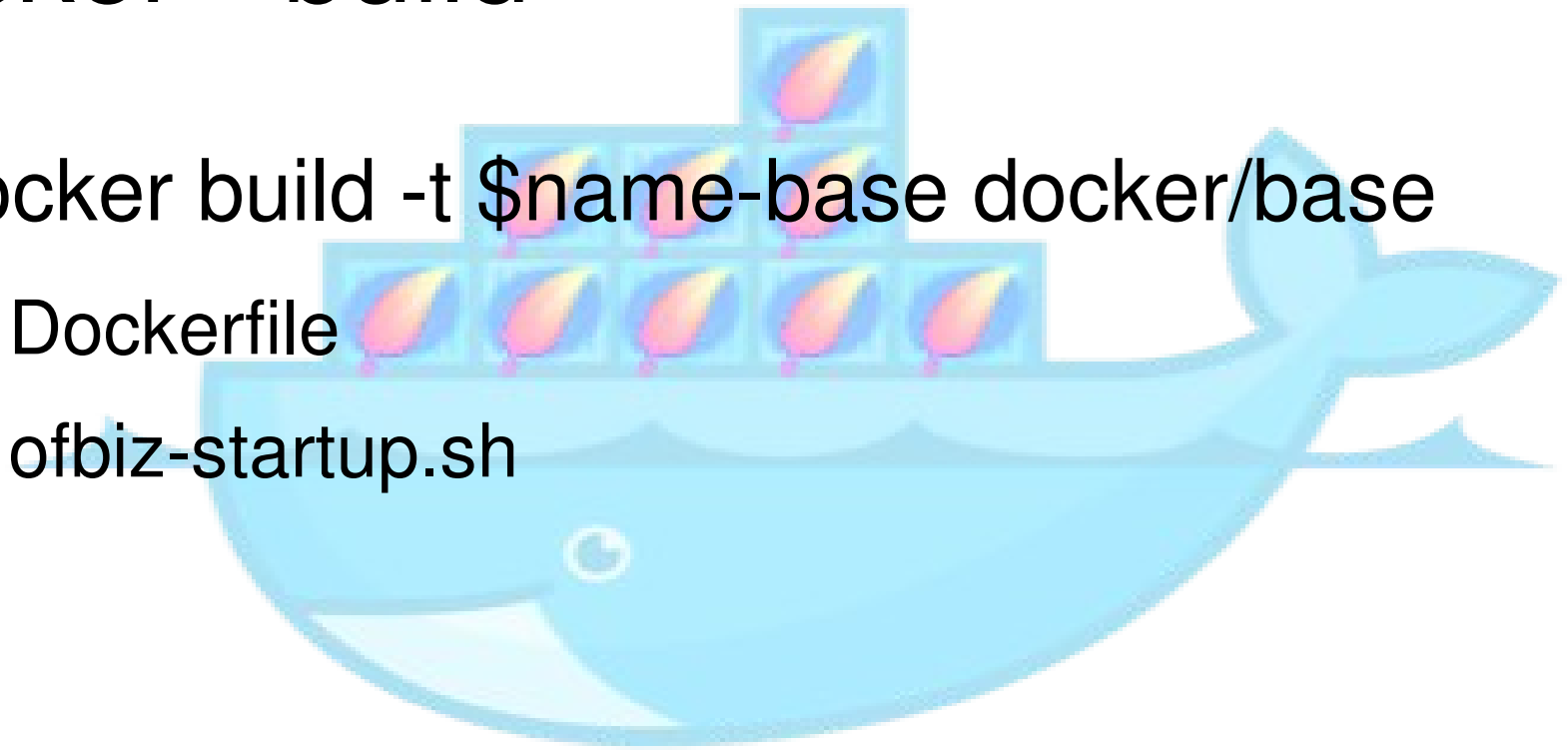
docker/base/Dockerfile

```
FROM debian:wheezy
# install packages
RUN apt-get install postgresql-9.1 libpostgresql-java nginx-full

# configure packages
USER root
RUN ln -s /srv/ofbiz/etc/nginx.conf /etc/nginx/sites-enabled/ofbiz.conf
USER postgresql
RUN createdb ofbiz; create user ofbiz

# this will start postgres, ofbiz, and nginx
COPY ofbiz-startup.sh /etc/init.d/ofbiz-startup
ENTRYPOINT /etc/init.d/ofbiz-startup
```

docker - build

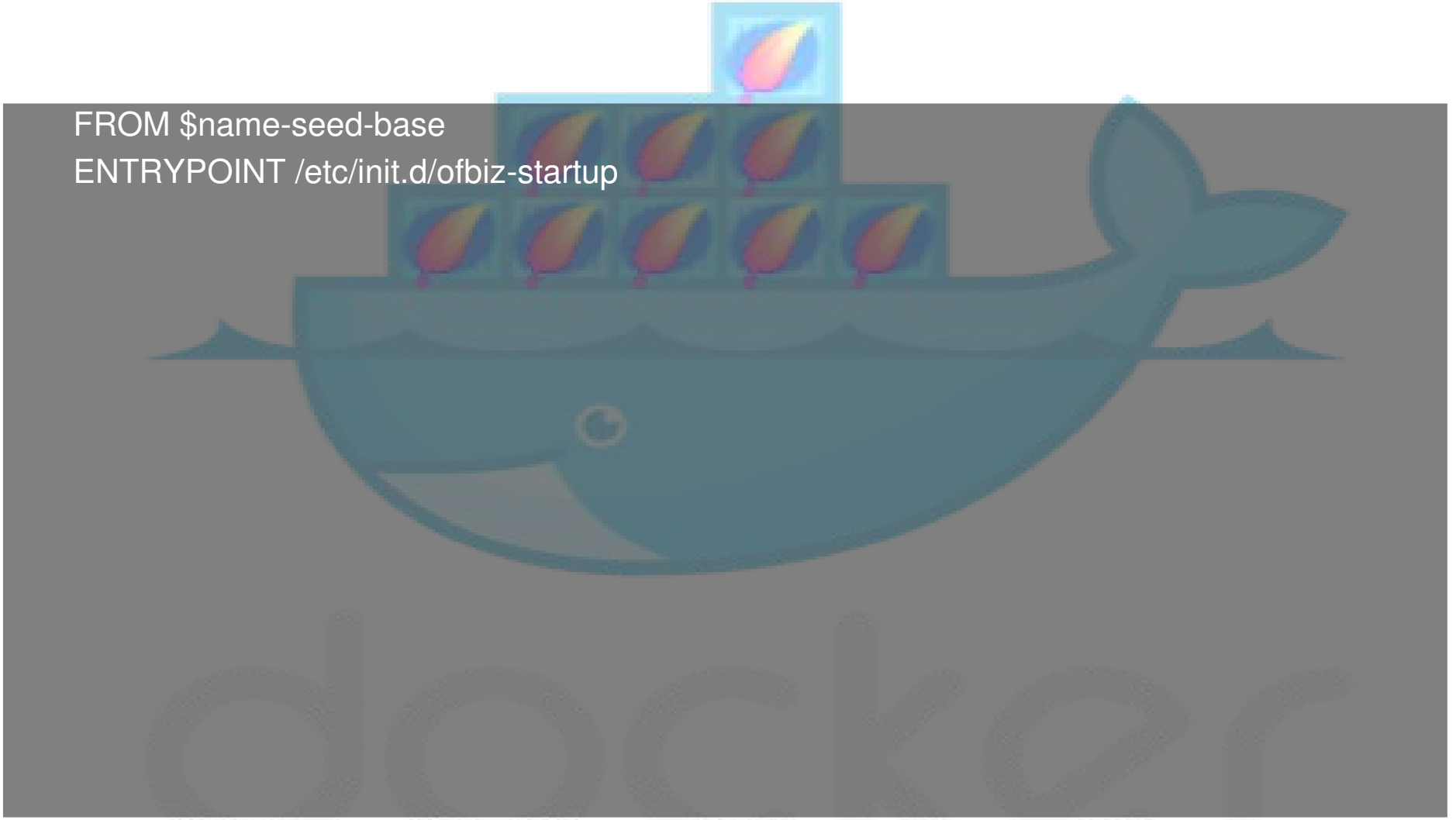


- docker build -t $name-base docker/base
  - Dockerfile
  - ofbiz-startup.sh

docker - seed

```
$ id=$(docker create -v $PWD:/srv/ofbiz $name-base)
$ docker start $id
$ # wait
$ docker exec $id “/srv/ofbiz/ant load-seed”
$ docker stop $id
$ docker commit $id $name-seed-base
```

dockerfile - seed



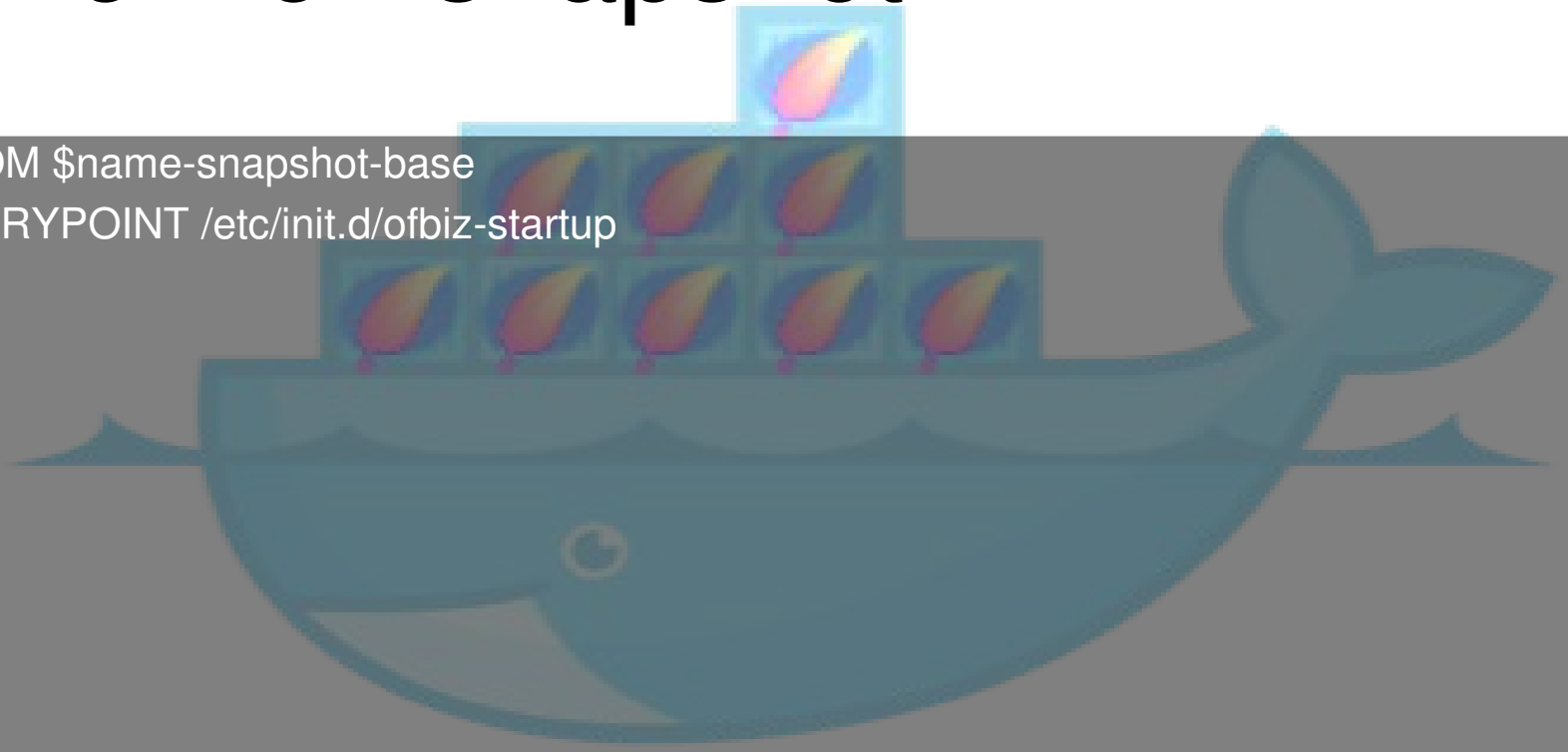
```
FROM $name-seed-base
ENTRYPOINT /etc/init.d/ofbiz-startup
```

docker - snapshot

```
$ id=$(docker create -v $PWD:/srv/ofbiz $name-base)
$ docker start $id
$ # wait
$ docker exec $id “zcat /srv/ofbiz/dumps/pgdump.sql.gz | su - postgresql”
$ docker stop $id
$ docker commit $id $name-snapshot-base
```

dockerfile - snapshot

```
FROM $name-snapshot-base
ENTRYPOINT /etc/init.d/ofbiz-startup
```



docker - run

- docker run -v $PWD:/srv/ofbiz $name
  - Created write-layer on top of files in $name
  - Mounts the current directory at the requested location
  - Runs the previously defined ENTRYPOINT