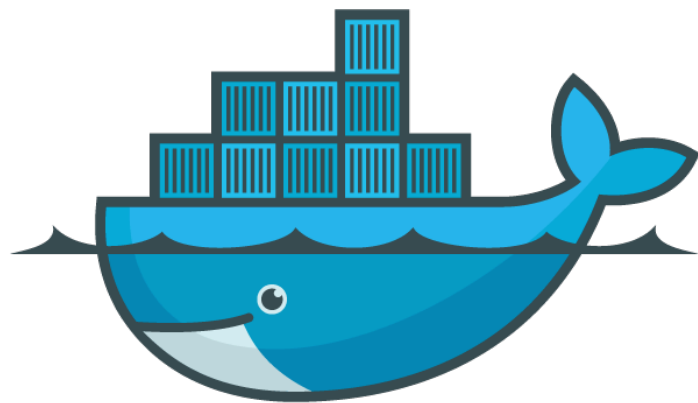


Docker

In the **ORACLE** Universe

OTN Tour South America / August 2106

Dr. Frank Munz



docker

munz & more 



... some basics



#OOV2014 ... Docker?"

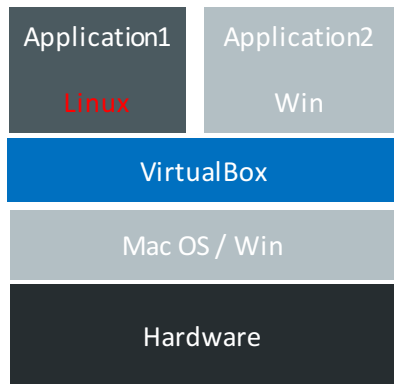
Docker

- Open Source (evolving), written in Go
- Container technology
- Portable standard
- Runs on Linux (Microsoft, MacOS, Solaris)

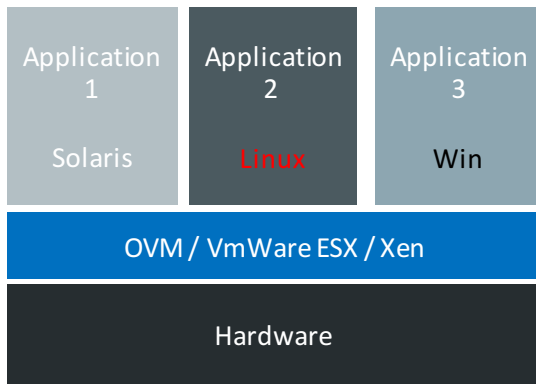


Google starts
2.000.000.000
containers
per week!

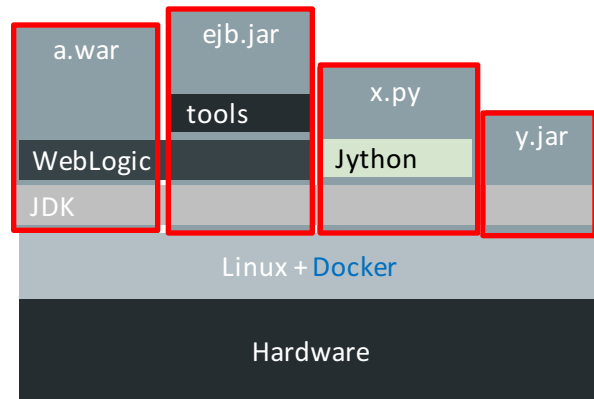
Virtualization vs. Isolation



Desktop Virtualization:
type 2 hypervisor
= with host OS

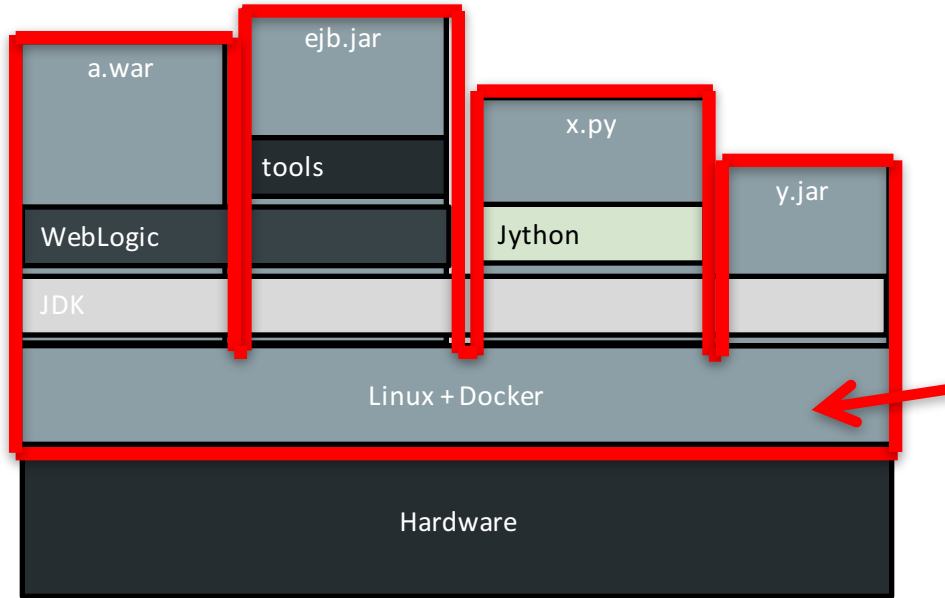


Server Virtualization
type 1 hypervisor
= on bare metal



Docker **container** in Linux
with own FS, network stack /
IP address, process space and
resource limits
-> Isolation

Docker



Docker is not a lightweight VirtualBox - it's about **isolation**.

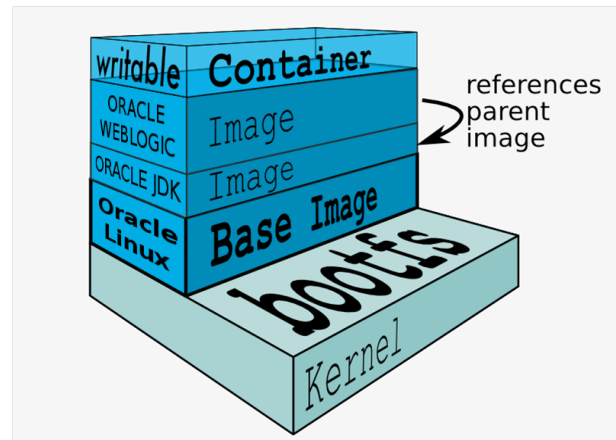
Containers run on Linux kernel of host

-> Containers are visible on host

Docker Images

- Package format
- Layered incremental, copy on write file system
- "Application with all dependencies"
- Create image yourself or get it from Docker Hub

docker images



Example Layers:

- WLS Domain
- WebLogic
- Java
- Base Image

Docker Container

- Isolated runtime of Docker image
- Starts up in milliseconds
- Sandboxing uses Linux namespaces and cgroups (RAM, CPU, filesystem)
-> isolated part of your Linux
- Open Container Standard / Linux Foundation

```
docker run -d -p 8080:9999 fmunz/micro
```

Docker Limitations

- Cannot load kernel modules
- Applications that manipulate namespaces
- Kernel config per container
- Some SW not (yet) supported when running in Docker container: Oracle DB etc.

Solves the “Worked For Me!” issue

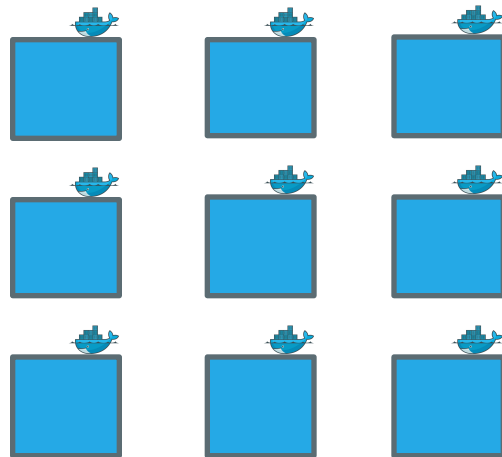
OS utils, JDK, patches, database driver, libs,
appserver, domain, **deployment**, tools,
scripts

dockerize it!

Docker 



Production



You can pass environment variables
for specific settings e.g. in prod

And Now Automate


- Build Docker images for testing in continuous delivery pipeline
- Use Jenkins / Hudson hooks or a maven plugin to create / start / stop / delete Docker containers

... automate, automate, automate

Various maven plugins available, e.g.

R. Huss (Jolokia REST-JMX bridge):

<https://github.com/rhuss/docker-maven-plugin>



	wouterd	alexc	spotify	rhuss
API	jaxrs	docker-java (forked)	spotify/docker-client	UniREST
Start/Stop	✓	✓	✗	✓
Building	✓	✓	✓	✓
Data Image	Dockerfile + Maven config	Dockerfile + custom YML	Maven config	Maven config + Assembly Descriptor
Push	✓	✓	✓	✓

	wouterd	alexc	spotify	rhuss
Cleanup	✓		✗	✓
Security	Plain (pom.xml, sys-props)	Plain (pom.xml, sys-props)	✗	Encrypted/Plain (settings.xml, pom.xml, sys-props)
URL Wait	✗	✓	✗	✓
Version	1.5	1.3.1	0.0.19-SNAPSHOT	0.9.8
Size	72k	21k	30k	63k

Dockerfile

Dockerfile

Manually create container with
docker build

GitHub



+



Automatic build



Docker Image



Dockerfile

```
33 # Pull base image
34 # -----
35 FROM oraclelinux:7
36
37 # Maintainer
38 # -----
39 MAINTAINER Bruno Borges <bruno.borges@oracle.com>
40
41 # Environment variables required for this build (do NOT change)
42 # -----
43 ENV JAVA_RPM jdk-7u79-linux-x64.rpm
44 ENV WLS_PKG fmw_12.1.3.0.0_wls.jar
45 ENV JAVA_HOME /usr/java/default
46 ENV CONFIG_JVM_ARGS -Djava.security.egd=file:/dev/./urandom
47
48 # Setup required packages (unzip), filesystem, and oracle user
49 # -----
50 RUN mkdir /u01 && \
51     chmod a+rx /u01 && \
52     useradd -b /u01 -m -s /bin/bash oracle
53
54 # Copy packages
55 COPY $WLS_PKG /u01/
56 COPY $JAVA_RPM /u01/
57 COPY install.file /u01/
58 COPY oraInst.loc /u01/
59
60 # Install and configure Oracle JDK 8u25
61 # -----
62 RUN rpm -i /u01/$JAVA_RPM && \
63     rm /u01/$JAVA_RPM
64
65 # Change the open file limits in /etc/security/limits.conf
66 RUN sed -i '/*EOF/d' /etc/security/limits.conf && \
67     echo "* soft nofile 16384" >> /etc/security/limits.conf && \
68     echo "* hard nofile 16384" >> /etc/security/limits.conf && \
69     echo "/*EOF" >> /etc/security/limits.conf
70
```

Manually create container:

`docker build -t name .`

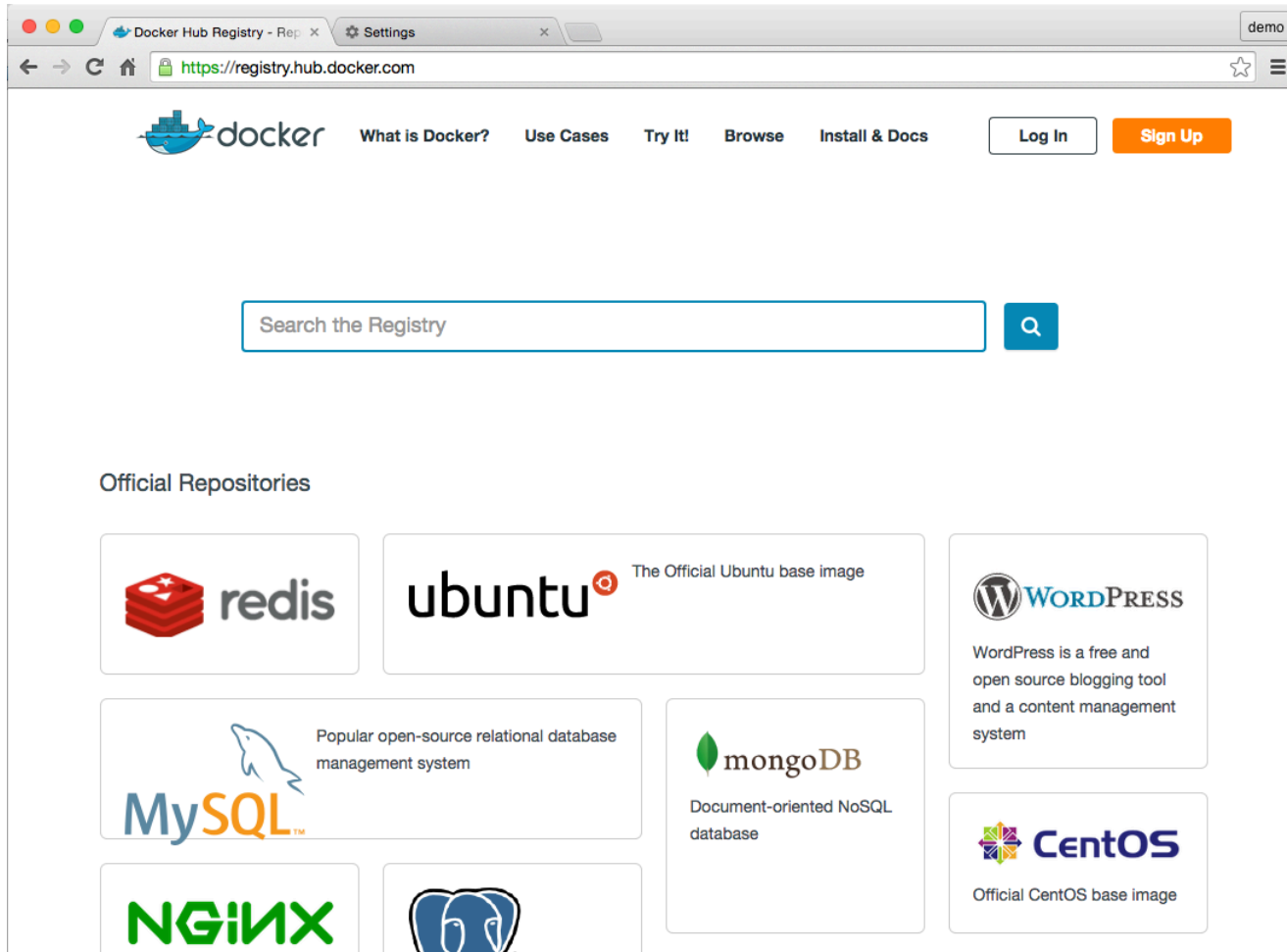
the registry

Registry

Hosted, code open sourced

- Docker image is not found? pulled from registry
- Push your image to registry
docker push yourname/newimage
- Free account includes 1 private registry

Also private, [containerized](#) registry for download with fs and optional in-memory, S3, or Azure data store



Docker Registry

what should be your
biggest nightmare:

unknown and
unofficial images
(>14000)

Automated Builds

- Automatically build your images: GitHub account with Dockerfile
 - Registry uses GitHub directory structure as build context
 - Image is uploaded automatically to Docker hub
- > Trust, up to date, and transparent





clouds

Docker in the Cloud?

Supported by every major cloud provider:



On premise -> **all** clouds

Oracle Cloud and Docker

Docker Container Service (announced)

- Expectation: you can run your Docker containers and orchestrate them

Application Container Cloud Service

- Uses Docker containers to run your Java or JavaScript application

Compute Cloud Service

- Manually run your containers

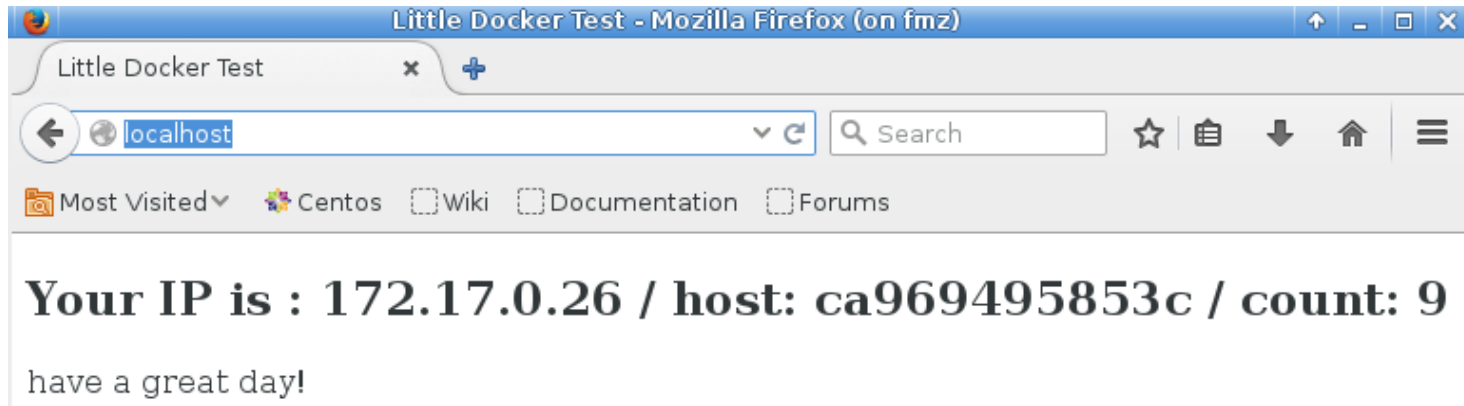
demo?

Small Images / Microservices

You can have a real service in ...

```
$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	VIRTUAL SIZE
oracle/dom1	latest	bc2020c6bd0f	3 hours ago	2.027 GB
fmdom1	latest	bc2020c6bd0f	3 hours ago	2.027 GB
oracle/weblogic	12.1.3-dev	2aa8eadf6c86	3 hours ago	2.025 GB
micro	latest	dfdcdb33a11fe	6 hours ago	8.488 MB
ubuntu	latest	8251da35e7a7	4 days ago	188.4 MB
centos	latest	7322fbe74aa5	7 weeks ago	172.2 MB
oraclelinux	7	8a2b759d9dd8	8 weeks ago	189.6 MB



Possible
Options:
busybox and
static binary

Simple Life Inside Container

```
Mem: 5023576K used, 2950096K free, 0K shrd, 0K buff, 140346646331393K cached
CPU:  0% usr  0% sys  0% nic 100% idle  0% io  0% irq  0% sirq
Load average: 0.12 0.19 0.13 2/1612 13
```

PID	PPID	USER	STAT	VSZ	%VSZ	%CPU	COMMAND
1	0	root	S	7192	0%	0%	python /webserver.py
9	0	root	S	3176	0%	0%	/bin/sh
13	9	root	R	3168	0%	0%	top

processes

/ #

/ # ls

bin	etc	lib	linuxrc	mnt	proc	run	sys	usr	webserver.py
dev	home	lib64	media	opt	root	sbin	tmp	var	
/ # █									

FS

/ # df

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
none	19049892	2568556	15490612	14%	/
tmpfs	3986836	0	3986836	0%	/dev
shm	65536	0	65536	0%	/dev/shm
/dev/sda1	19049892	2568556	15490612	14%	/etc/resolv.conf
/dev/sda1	19049892	2568556	15490612	14%	/etc/hostname
/dev/sda1	19049892	2568556	15490612	14%	/etc/hosts
tmpfs	3986836	0	3986836	0%	/proc/kcore
tmpfs	3986836	0	3986836	0%	/proc/timer_stats

mounts

#3

Security

```
$ docker run -d -p  
8080:9999 fmunz/micro
```

VS.

a complete stranger gives you a
box at night and asks you to
connect it to your company
network:



Would you do it?

Suggestions

- Use trusted images / with known Dockerfile
- Kernel features are well established
 - cgroups (2006, merged into 2.6.24 kernel)
 - namespaces (initial kernel patch 2.4.19)
- Docker can use TLS (client to daemon)
- Docker images can be signed
- Think about pulling images from public repos / Docker hub



FUD

"Docker is like chroot() on steroids."

- Yes: It's easy to escape chroot() environment
- No: Docker does **not** use chroot()
-> it uses namespaces

Do namespaces solve it?

6 different namespace, but
not everything is namespaced, eg:

- /proc/sys | irq | bus
- /sys, /sys/fs
- /dev/mem
- /dev/sd*
- kernel modules
- No user namespaces (but experimental in 1.9)

Docker uses read-only mounts where possible

Linux Capabilities

- Privileged container: like having root on host
- Capabilities -> Break down power of root
- Examine PID 1 capabilities with getpcaps:

```
$ getpcaps 1
Capabilities for `1': = cap_chown,cap_dac_override,cap_dac_read_search,cap_fowner,cap_fsetid,cap_kill,
cap_setgid,cap_setuid,cap_setpcap,cap_linux_immutable,cap_net_bind_service,cap_net_broadcast,cap_net_a
dmin,cap_net_raw,cap_ipc_lock,cap_ipc_owner,cap_sys_module,cap_sys_rawio,cap_sys_chroot,cap_sys_ptrace
,cap_sys_pacct,cap_sys_admin,cap_sys_boot,cap_sys_nice,cap_sys_resource,cap_sys_time,cap_sys_tty_conf
ig,cap_mknod,cap_lease,cap_audit_write,cap_audit_control,cap_setfcap,cap_mac_override,cap_mac_admin,cap
_syslog,35,36+ep
ccloud:/home/frank
$
ccloud:/home/frank
$ docker run -it centos getpcaps 1
Capabilities for `1': = cap_chown,cap_dac_override,cap_fowner,cap_fsetid,cap_kill,cap_setgid,cap_setui
d,cap_setpcap,cap_net_bind_service,cap_net_raw,cap_sys_chroot,cap_mknod,cap_audit_write,cap_setfcap+ei
p
```


"Containers don't contain!"

Quote by D. Walsh, Mr. SE Linux <- !!

SELinux = what a process is able to do based on rules.

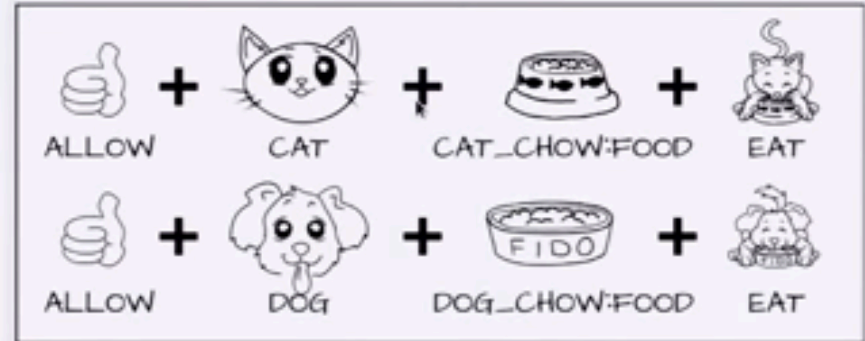
Enforcement:

A really bad idea:
setenforce 0

containerProcessType
can only read/exec
/user files

and only write to
containerFileType

TYPE ENFORCEMENT



... more Suggestions

- Drop privileges as quickly as possible
- Treat root in container as root outside (although it isn't)
- No secrets in images
- Combine Docker with SELinux, AppArmor and / or virtualization
- Host can always access container

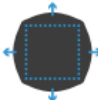
Note: Public PaaS do not simply spin up Docker containers!

Cheat Sheet

TYPES OF SECURITY THREATS AND HOW TO AVOID THEM



KERNEL EXPLOITS
If a container can cause a kernel panic or similar, it will bring down the whole host.



DENIAL OF SERVICE (DOS) ATTACKS
All containers share kernel resources. If one container monopolizes access to a resource, it will starve out the other containers.



CONTAINER BREAKOUTS
If an attacker can breakout of a container, they can gain access to the host and other containers.



POISONED IMAGES
Images may be injected with trojan or virus infected software. Or they may simply be running outdated, known-vulnerable versions of software.



COMPROMISED SECRETS
API keys and database passwords must be kept secure to prevent attackers gaining access.

SEGREGATE CONTAINER GROUPS WITH VMs		○			
DEFANG SETUID/SETGID BINARIES	○		○		
BE AWARE OF CPU SHARES		○			
VERIFY IMAGES				○	
SET CONTAINER FILE SYSTEM TO READ-ONLY	○	○	○		○
SET A USER	○		○		○
DO NOT USE ENVIRONMENT VARIABLES TO SHARE SECRETS					○
DO NOT RUN CONTAINERS WITH THE --privileged FLAG	○		○		○
TURN OFF INTER-CONTAINER COMMUNICATION	○	○	○		
SET VOLUMES TO READ-ONLY	○		○		
SET MEMORY LIMITS		○			
DO NOT INSTALL UNNECESSARY PACKAGES IN THE CONTAINER	○		○		

Source: Container-Solutions.com

User Namespaces

Docker 1.9 experimental supports user namespaces:

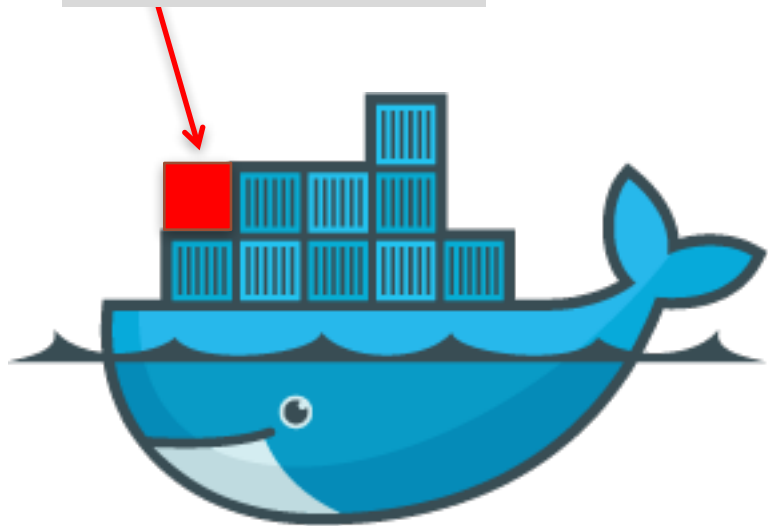
root in container != root on host

Conclusion

- You have to deal with Docker security depending on your use case
- Note: Public PaaS are not just spinning up Docker containers they use SELinux, VMs,...
- Docker is not a risk per se but new technology with different challenges.

Docker in Production?

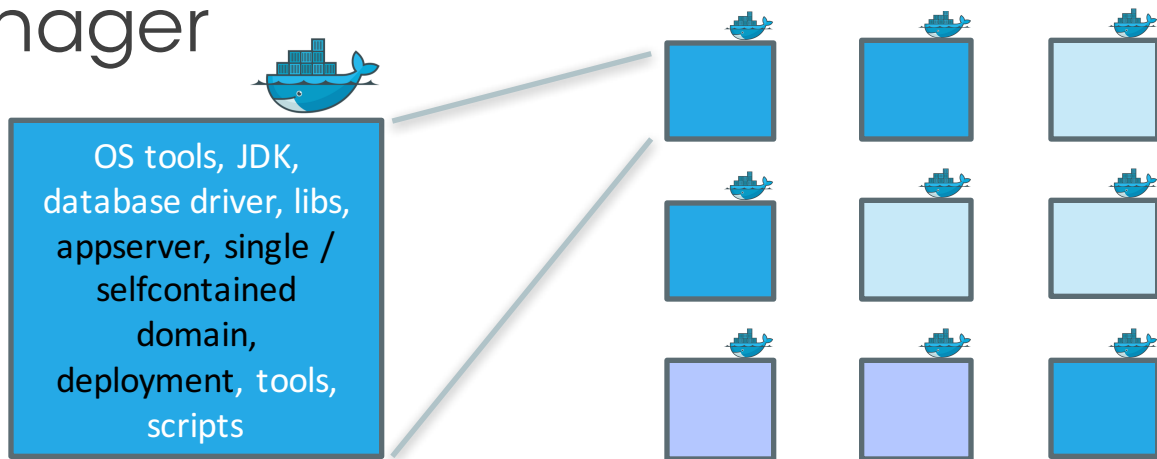
WebLogic
in a Docker
Container!



docker

Docker Style


- Independent appserver in container
- Microservices style architecture
- Just add your favorite Docker cluster manager



Links (OLD): WebLogic Example

```
$docker run -d -p 8001:8001  
--name=wlsadmin  
fmdom1  
startWebLogic.sh
```

```
$docker run -d  
--link wlsadmin:wlsadmin  
fmdom1  
createServer.sh
```

port 8001 

OLD STYLE!
Use **Networks** now...

--link

IP:port 7001 

**connect to admin
due to --link:
/etc/hosts**

172.17.1.99 wlsadmin 31a1baaf

Managed Servers

Oracle

Oracle Product in Docker	Official Support
GlassFish	
MySQL	yes
NoSQL	
OpenJDK	
Oracle Linux	yes
OracleCoherence	yes
OracleDatabase	no
OracleHTTPServer	yes
OracleJDK	yes
OracleTuxedo	yes
OracleWebLogic	yes

Oracle support
does **not require**
you to use the
provided Docker
files

https://github.com/oracle/docker-images

GitHub - oracle/docker-images

https://github.com/oracle/docker-images

Apps P WLS DOAGC OUGN WLS Console DockerAdminWLS Docker Hub Little Docker Test etc OOW2015 OSB12.2.1

oracle / docker-images

Watch 110 Star 421 Fork 262

Code Issues 0 Pull requests 0 Pulse Graphs

Official source for Docker configurations, images, and examples of Dockerfiles for Oracle products and projects

490 commits 3 branches 0 releases 23 contributors

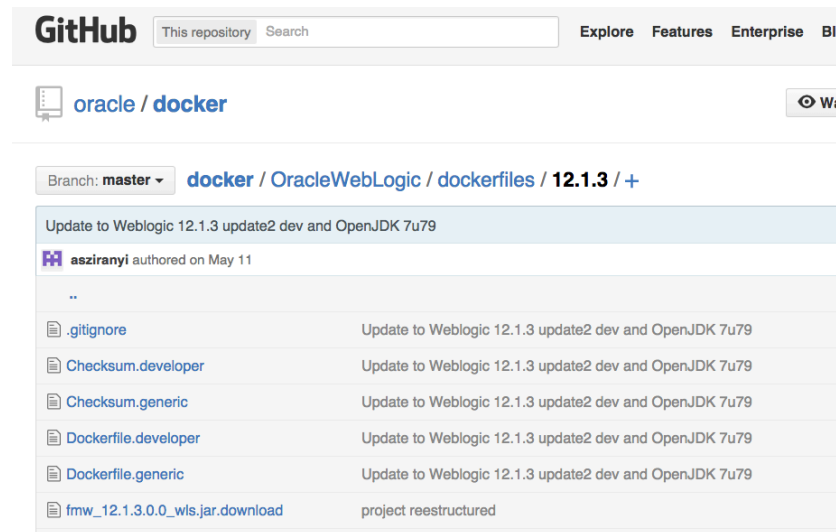
Branch: master New pull request Find file Clone or download

gvenzl committed with brunoborges New samples and minor bug fixes (#130) Latest commit bcd2cfb 2 days ago

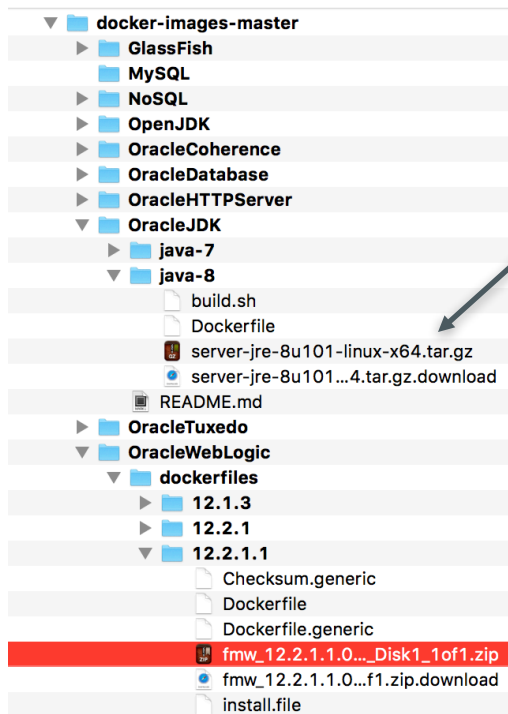
GlassFish	references oraclelinux from top-level repository. installs openjdk di...	6 months ago
MySQL @ 1cbaef6	pulled latest commit of MySQL submodule	8 months ago
NoSQL	removed jvm flag. moved var upper	6 months ago
OpenJDK	no need to do a clean all if going to delete /var/cache/yum/	6 months ago
OracleCoherence	Coherence Changes (#103)	2 months ago
OracleDatabase	New samples and minor bug fixes (#130)	2 days ago
OracleHTTPServer	Enhancement for OHS readme(s). (#115)	a month ago
OracleJDK	updated jdk8 to update 101	8 days ago
OracleTuxedo	merged changes for Tuxedo	2 months ago
OracleWebLogic	new 12.2.1.1 dockerfile. generic distro only for now	29 days ago

WebLogic: What Do You Get?

- NOT WebLogic from Docker registry
- NO automatic build via github
- Github repo with scripts to set up WebLogic on Oracle Linux in Docker
- Generic distribution
- Docker is a supported environment for WebLogic 12.1.3+



Just Drop Server JRE and WLS Installer



```
$ cd java-8
```

```
$ docker build -t oracle/jdk:8 .
```

```
Sending build context to Docker daemon 4.096 kB
```

```
Step 1 : FROM oraclelinux:latest
```

```
latest: Pulling from library/oraclelinux
```

```
10ec637c060c: Downloading 4.865 MB/97.84 MB
```

```
...
```

```
$ sh buildDockerImage.sh -g -v 12.2.1.1
```

```
...
```

Dockerfile

Dockerfile and Scripts
(from [Oracle github](#))

GitHub 

```
38 FROM oracle/jdk:8
39
40 # Maintainer
41 # -----
42 MAINTAINER Bruno Borges <bruno.borges@oracle.com>
43
44 # Environment variables required for this build (do NOT change)
45 # -----
46 ENV FMW_PKG=fmw_12.2.1.1.0_wls_Disk1_1of1.zip \
47     FMW_JAR=fmw_12.2.1.1.0_wls.jar \
48     ORACLE_HOME=/u01/oracle \
49     USER_MEM_ARGS="-Djava.security.egd=file:/dev/./urandom" \
50     PATH=$PATH:/usr/java/default/bin:/u01/oracle/oracle_common/common/bin
51
52 # Copy packages
53 # -----
54 COPY $FMW_PKG install.file oraInst.loc /u01/
55
56 # Setup filesystem and oracle user
57 # Install and configure Oracle JDK
58 # Adjust file permissions, go to /u01 as user 'oracle' to proceed with WLS installation
59 # -----
60 RUN chmod a+rx /u01 && \
61     useradd -b /u01 -m -s /bin/bash oracle && \
62     echo oracle:oracle | chpasswd && \
63     cd /u01 && $JAVA_HOME/bin/jar xf /u01/$FMW_PKG && cd - && \
64     su -c "$JAVA_HOME/bin/java -jar /u01/$FMW_JAR -silent -responseFile /u01/install.file -invPtrLoc /u01/
65     oraInst.loc -jreLoc $JAVA_HOME -ignoreSysPrereqs -force -novalidation ORACLE_HOME=$ORACLE_HOME
66     INSTALL_TYPE=\"WebLogic Server\" - oracle && \
67     chown oracle:oracle -R /u01 && \
68     rm /u01/$FMW_JAR /u01/$FMW_PKG /u01/oraInst.loc /u01/install.file
```



**\$docker build
-t wls:latest .**



WebLogic
Docker Image
(no domain)

Dockerfile

Example Dockerfile:

fmunz/supersmall

```
1 FROM busybox
2 ENV CITY Munich
3 CMD echo Hello $CITY today is `date`
```

Example Dockerfile:

hello-world:

```
1 FROM scratch
2 COPY hello /
3 CMD ["/hello"]
```



Extend the WLS-only image

Sample script provided:

- Dockerfile to extend WLS image
- Run WLST script to create domain
- Create boot.properties
- Expose NM, Server ports

WLS Domain Image

WebLogic Image

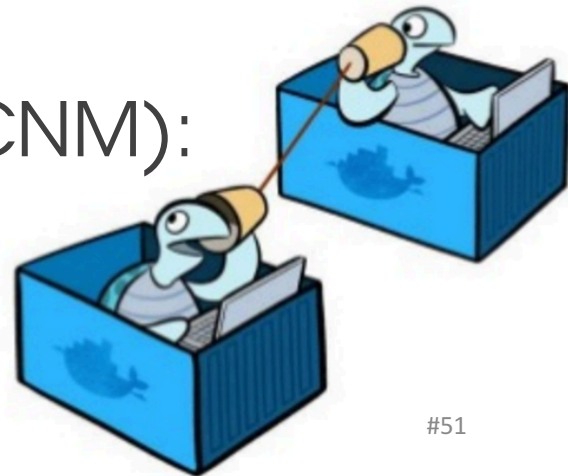
JDK Image

Linux Base Image

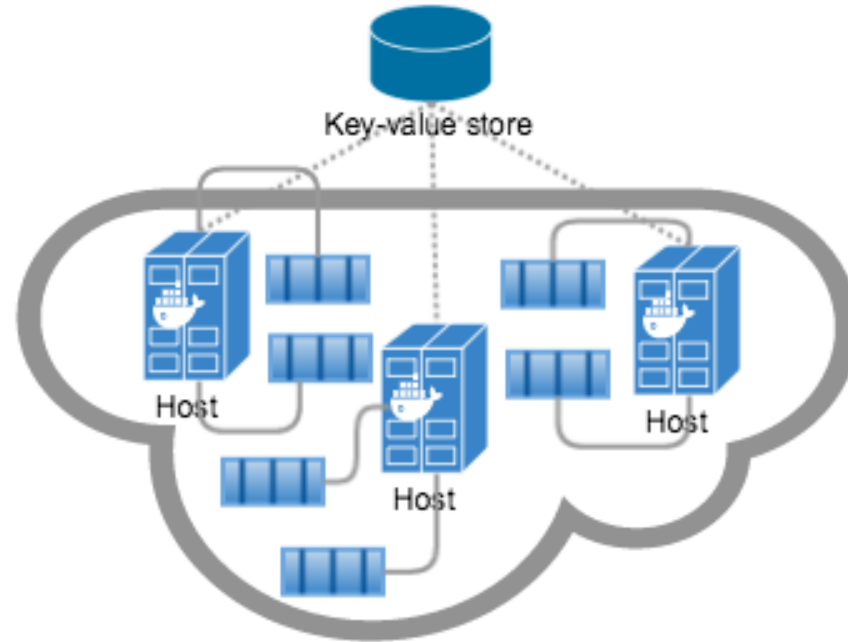
Docker Networking

Networking: Facts to Know

- Docker `--link` only works on single host
-> regarded as deprecated now
- Networking supported since Docker 1.9
- SDN network that spans hosts:
Libnetwork implements
Container Networking Model (CNM):
Endpoint / Network / Sandbox



Overlay Network



```
docker network create -d overlay
```

OracleWebLogic/samples/1221-multihost:

```
38 # Create overlay Docker Multihost Network and set Docker environment pointing to Machine
39 eval "$(docker-machine env --swarm $prefix-master)"
40 echo "Creating the Docker Network Overlay '$network' ..."
41 docker network create --driver overlay $network
42
```

Networking

- etcd, consul, or zookeeper used for machine discovery and meta data
- Top level API:
docker network
- Libnetwork, open sourced 04/2015,
500 pull requests
- Dynamically (dis)connect to multiple NW

Networking

```
[ $ docker network --help
```

```
Usage:  docker network [OPTIONS] COMMAND [OPTIONS]
```

Commands:

inspect	Display detailed network information
ls	List all networks
rm	Remove a network
create	Create a network
connect	Connect container to a network
disconnect	Disconnect container from a network

Orchestration / Cluster Manager

Docker Swarm

- Native Docker cluster with same API as a single engine
- Fast provisioning, about 500 msec
- Scheduling: spread, binpack, rand
- Features are optional, you can continue use Kubernetes etc.
- No insecure mode 😊



Docker Swarm

Since Docker 1.12

- Swarm is merged with Docker engine:
- Load balancer included
- Service discovery
- Cluster scheduler

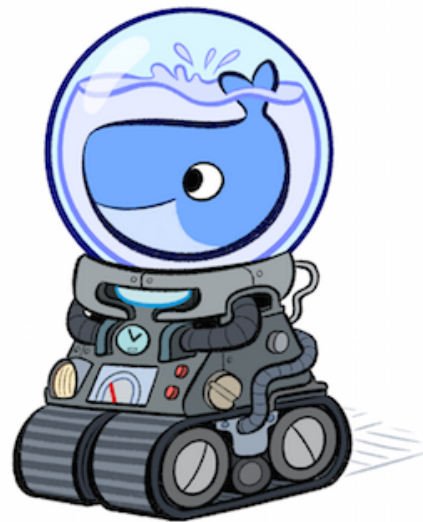
-> Swarm has more features like Google's Kubernetes
-> easier to get started

Docker Machine

- Provision Docker in VirtualBox, Vmware, GCE, AWS, DigitalOcean etc.

```
docker-machine \  
create -d=virtualbox default
```

- Mac OS's boot2docker is replaced by Docker Machine, which again is replaced by native Docker on Mac now



Updates Images?

You could use Docker copy command – yet it's not hip in the cloud to update. Just rebuild the container.



*"Servers are cattle.
Not pets."*

-> immutable server



Predictions

- Swarm will take its share from Kubernetes.
- You will **not** dockerize 90% of your enterprise IT in the next 24 months.
- **Docker is the new Linux.**
Be ready to experience that feeling we had with Linux 13 years ago 😊

Conclusion

- Docker is ready for prime time!
- Docker itself, but more so cluster managers are still evolving
- Docker itself is not a security risk, but make sure to tick off the security checklist
- Oracle caught the trend early – good!
- Some products supported, more to come?

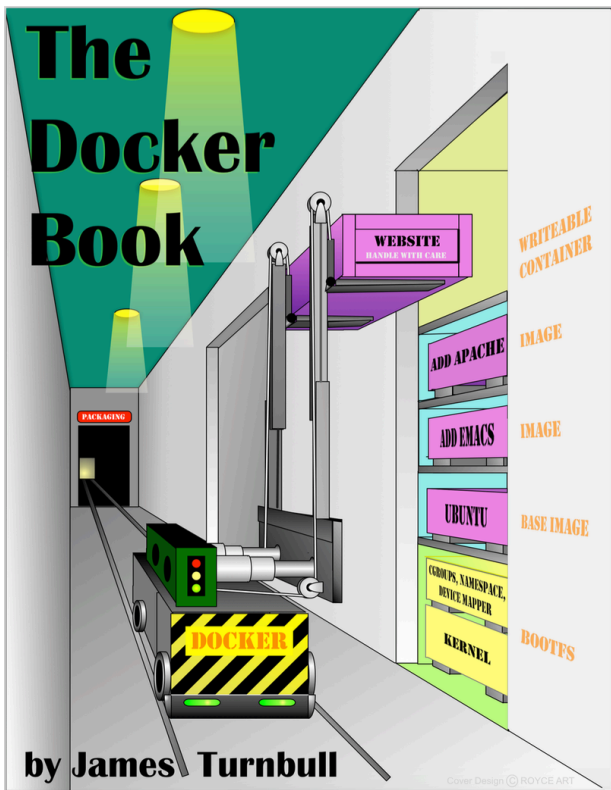
Oracle WebLogic Server on Docker Containers

ORACLE WHITE PAPER | OCTOBER 2015



Oracle Whitepaper WebLogic on Docker

<http://www.oracle.com/us/products/middleware/cloud-app-foundation/weblogic/weblogic-server-on-docker-wp-2742665.pdf>



THE DOCKER BOOK

CONTAINERIZATION IS THE NEW VIRTUALIZATION



SIMPLE

A hands-on book that teaches you Docker™.



SCALABLE

Start small with a single container and then build on what you learn to deploy multi-container applications.



UP-TO-DATE

Current, accurate and up-to-date.



DIFFERENT

Written for both developers and sysadmins with real-world examples and use cases.

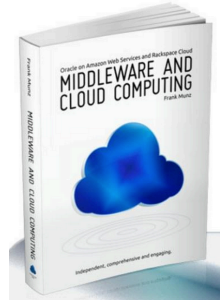
Good Docker book by
J. Turnbull
(covering Docker 1.10)

muito obrigado!

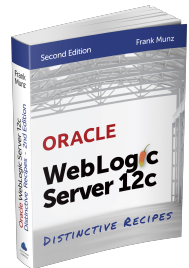
You can win a book

... if you promise to write a short
review on [Amazon.com](https://www.amazon.com)

tweet to win!



#otntourla **OR** @soacommunity
@frankmunz
+picture?



Don't be
shy 😊

 www.munzandmore.com/blog

 facebook.com/cloudcomputingbook

 facebook.com/weblogicbook

 @frankmunz

 youtube.com/weblogicbook
-> more than 50 web casts

