



Docker

Conteneurs et orchestrations





Docker FOr Mac

Oui c'est lent

à cause d'inconsistances dans la virtualisation du kernel , la virtualisation de la propagation d'events inotify

Docker utilise un osxfs volume partagé entre l'hôte et le xHyve



Docker-compose volume mount

1. **consistent**
2. **cached** — Updates may have a lag before showing up on the host side.
3. **delegated** — Updates may have a lag before showing up on the container side.

--v - my-code:/app/myapp:delegated



Solution 2 : DMS, Docker Sync, etc.

<https://github.com/mickaelperrin/docker-magic-sync> (26*)

<https://github.com/EugenMayer/docker-sync> (3,1k*)

docker-sync.yml

version: "2"

syncs:

appcode-native-osx-sync:

src: './app'

sync_excludes: ['ignored_folder', 'ignored_dot_folder']



Mutagen

sync:

defaults:

mode: "two-way-resolved"

flushOnCreate: true

ignore:

vcs: true

my-volume-name:

alpha: "php/src"

beta: "docker://php_example/app/src"

ignore:

paths:

- ".idea"
- "node_modules"
- "php/src/vendor"



<https://mutagen.io/>



12 factors apps

Twelve-Factor Applications

1. CODEBASE

One codebase tracked
in SCM, many deploy

2. DEPENDENCIES

Explicitly declare
isolate dependencies

3. CONFIGURATION

Store config in
the environment

4. BACKING SERVICES

Treat backing services as
attached resources

5. BUILD, RELEASE, RUN

Strictly separate build
and run stages

6. PROCESSES

Execute app as
stateless processes

7. PORT BINDING

Export services via
port binding

8. CONCURRENCY

Scale out via the
process model

9. DISPOSABILITY

Maximize robustness &
graceful shutdown

10. DEV/ PROD PARITY

Keep dev, staging, prod as
similar as possible

11. LOGS

Treat logs as
event stream

12. ADMIN PROCESSES

Run admin / mgmt tasks
as one-off processes



Merci !

