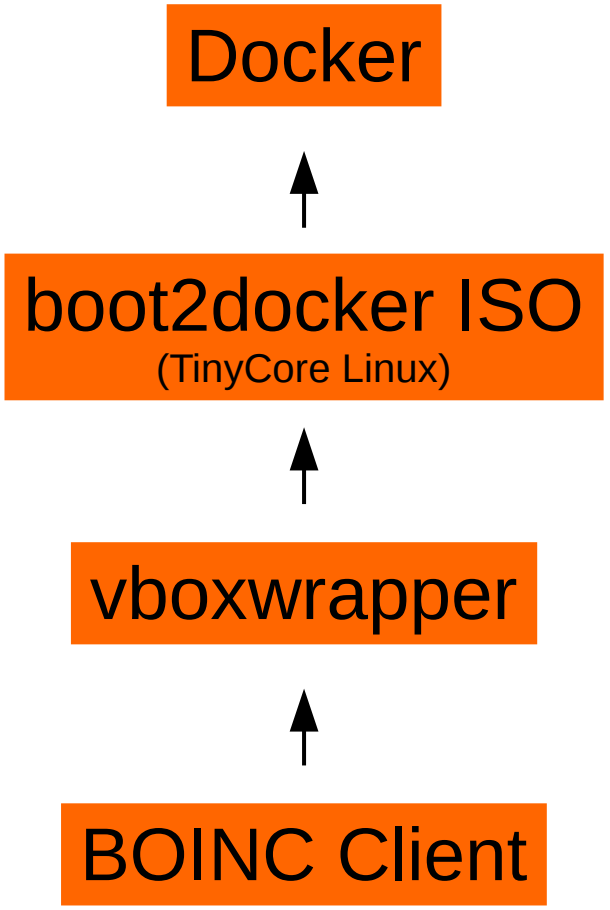
boinc2docker: creating apps so easy even a physicist can do it

Marius Millea

Cosmology@Home

demo3

Structure

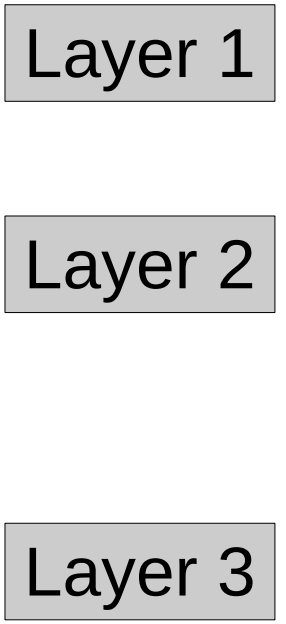

Docker
boot2docker ISO
(TinyCore Linux)
vboxwrapper
BOINC Client

Layers & Input files

The Cosmology@Home “camb_boinc2docker app”

Dockerfile

```
1  FROM debian:jessie
2
3  RUN apt-get update \
4      && apt-get -y install \
5          curl \
6          gfortran \
7          libgomp1 \
8          make
9
10 # install camb
11 RUN mkdir camb \
12     && curl -L https://github.com/marius311/camb/tarball/33558ec
13     && cd camb \
14     && F90CRLINK="" make camb
15
16 ENTRYPOINT ["/root/camb/camb"]
17
```



- Each layer becomes a “sticky” BOINC input file
- Client only ever downloads a layer once, which can be very efficient if your apps use shared base layers
- Sharing base layers is ubiquitous in Docker

File sizes

vboxwrapper	2MB
boot2docker.iso	39MB
debian:jessie base image	53MB
Your app	??

Making images small is also ubiquitous in Docker

- Multi-stage builds
- “slim-the-filesystem-down” https://github.com/marius311/stfd

- Pros
  - Ease, reproducibility
  - Code your app in Linux, automatically works on Linux/Mac/Windows volunteer computers
  - Automatic check-pointing (thanks to vboxwrapper)
  - Docker is high-profile
- Cons
  - Requires volunteer to have
    - Virtualbox
    - 64bit computer
    - VM Extensions enabled
  - Some boot-up time overhead
  - Can’t run on low-priority on Windows
  - No GPU computing