

Using containers on CÉCI clusters

Olivier Mattelaer
UCLouvain
CISM

What do I want to cover



- What is a container
 - ➔ Why it can be interesting for you?



- Singularity: Container for HPC
 - ➔ Features
 - ➔ Limitations

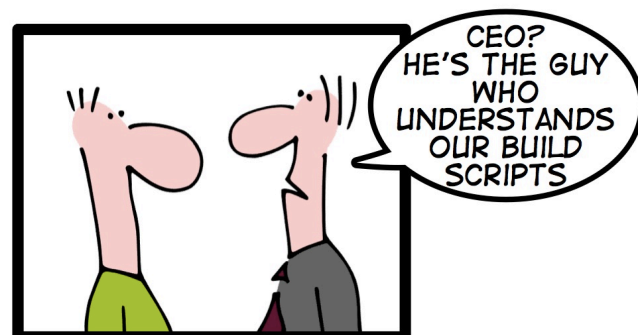
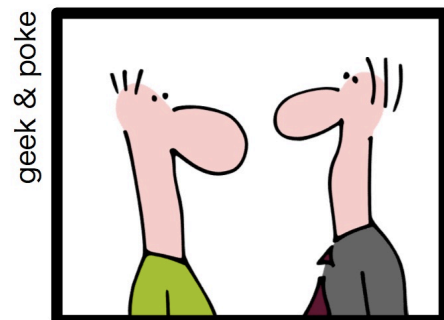
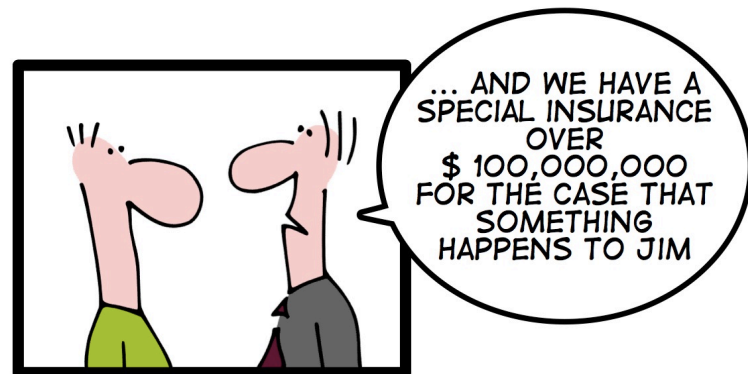


- Tutorial
 - ➔ Show that this is easy to do



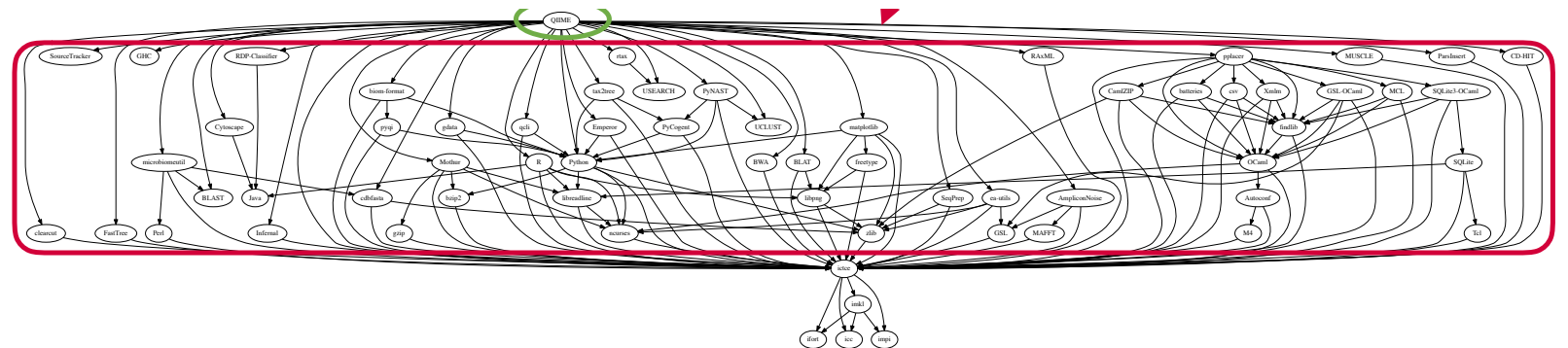
- HPC
 - ➔ Details on how to use our setup

Installing Software

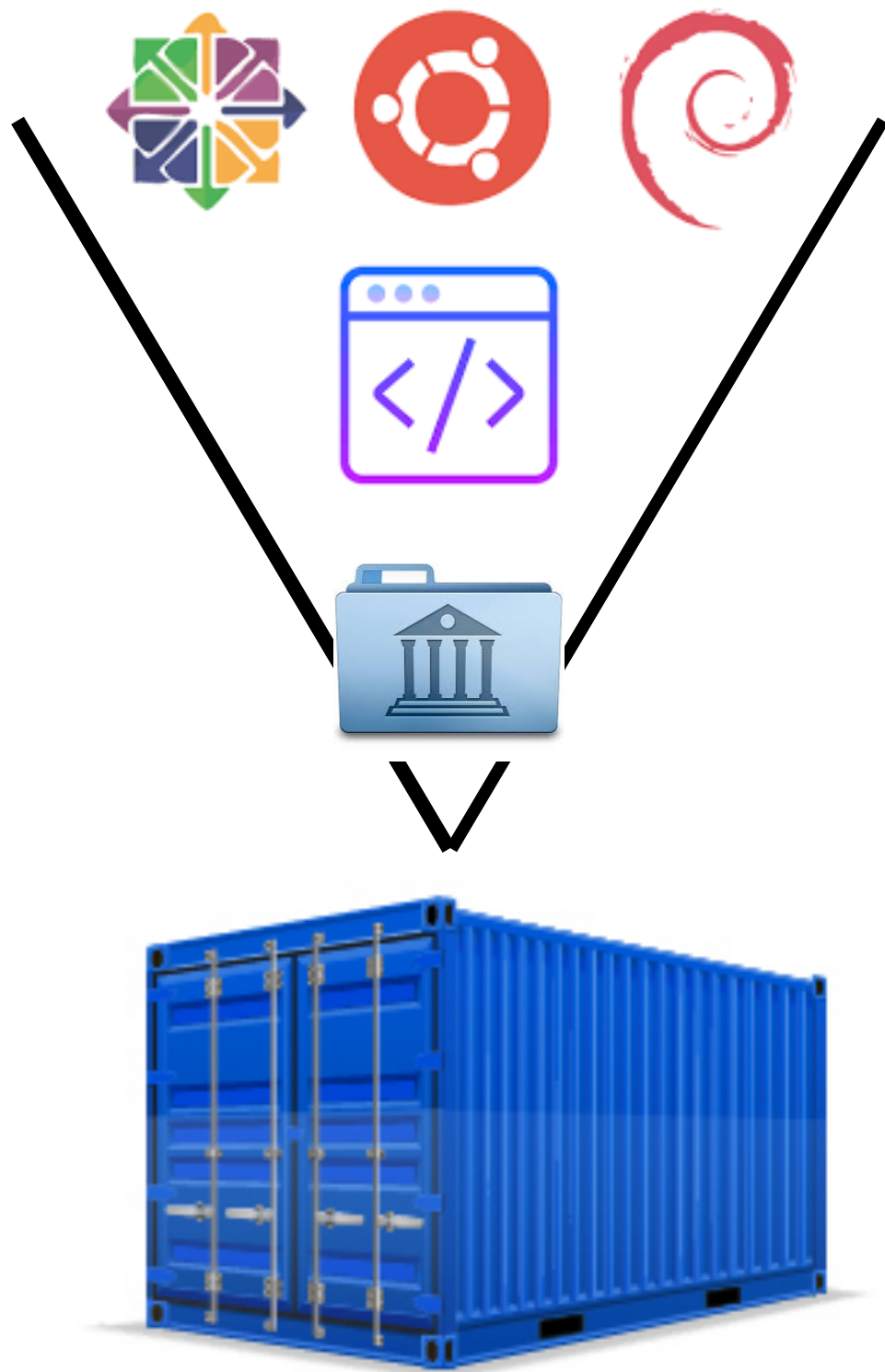


HOW TO BECOME INVALUABLE

- Tedious/complicated
 - ➔ For user
 - ➔ For sys-admin
- Dependencies Hell



Container Solution



- machine agnostic code
 - ➔ A (small) OS
 - ➔ Your code (executable)
 - ➔ All the dependencies (libraries)
- That can run “everywhere”

What for?



➔ **reproducibility** on any (unix) machine

✦ Nice to send to a collaborator !

➔ **deployment** (cloud/laptop/hpc/...)

✦ Nice to distribute the workload



➔ With a **paper**

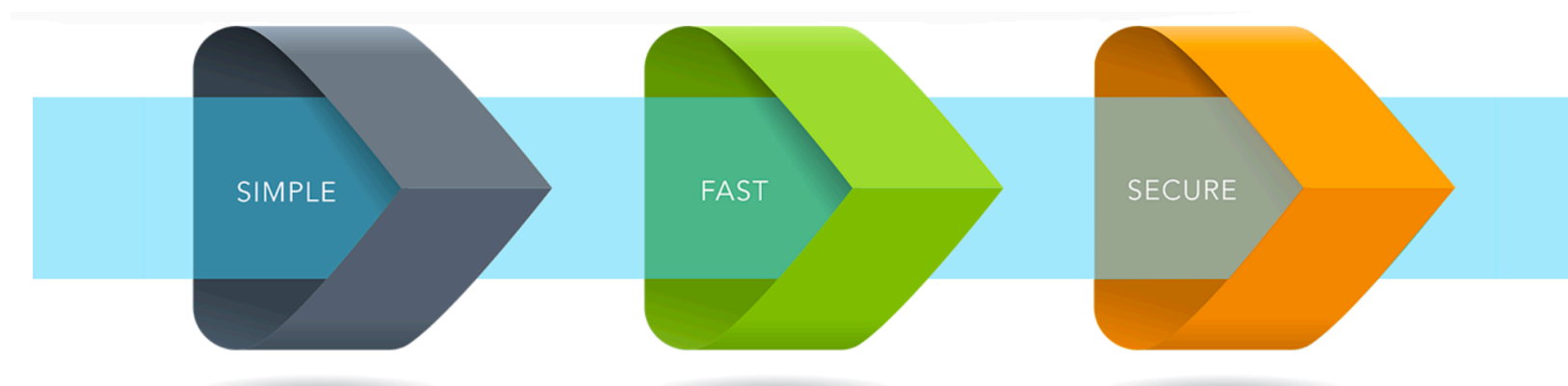
✦ Nice for being able to reproduce results

✦ Nice for other scientists

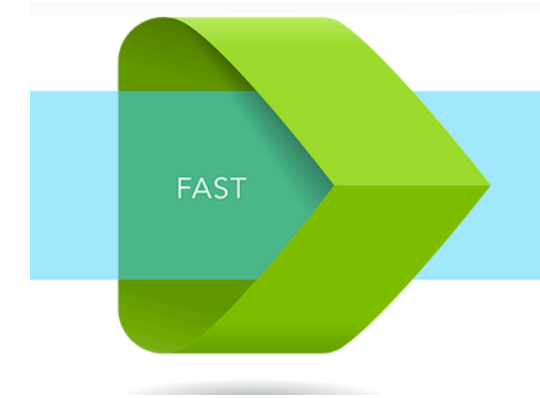


Containers History

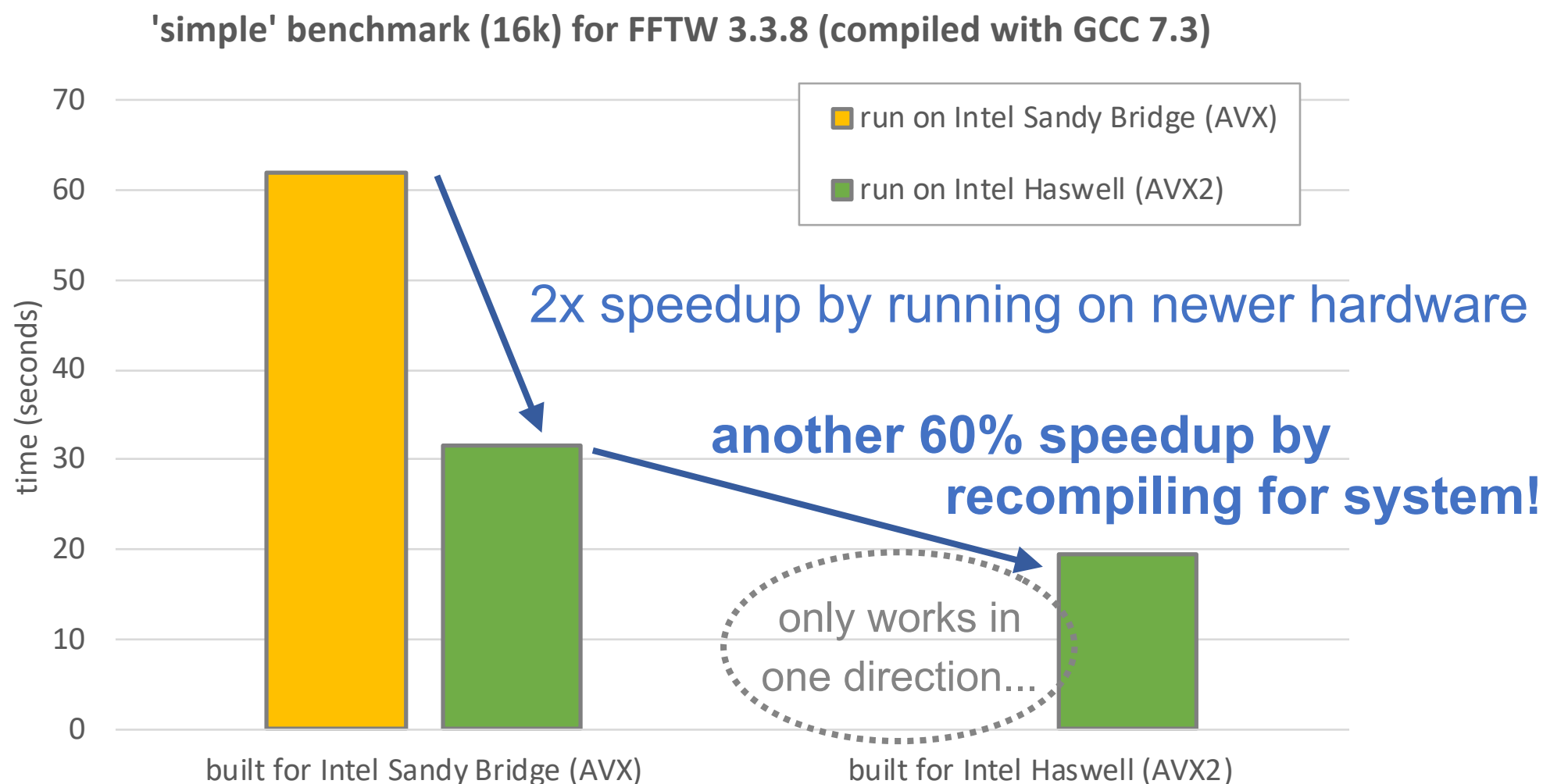
- Are an old idea
 - ➔ Chroot (1979), FreeBSD jails (2000), Solaris containers (2004), LXC (2008)
- Docker (2013)
 - ➔ For/with cloud computing
- Buzz for HPC containers starts ~ 2015
 - ➔ Docker tries to convince HPC structure and failed
- Singularity (2016)
 - ➔ HPC focus



Performance



- They claim “native” performance
 - ➔ understand “small” overhead (couple of percent)
 - ➔ No cpu optimisation



(FFTW 3.3.8 installed in Singularity container)

Plot taken from Kenneth Hoste

Hardware Optimisation

CPU



Need generic compilation

GPU



Special handling to handle GPU
Specific library at run time

MPI

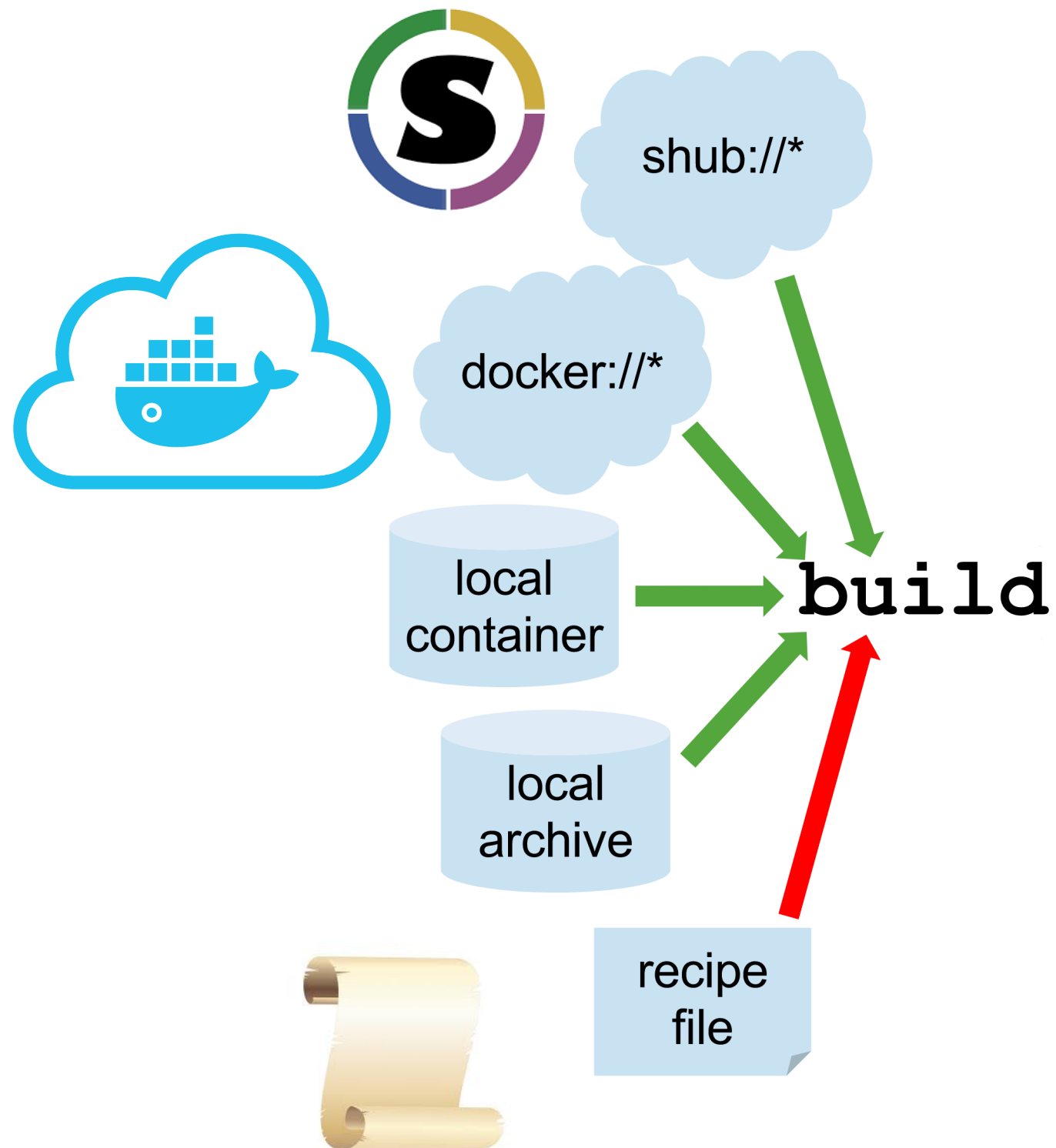


No special handling
But actually needed

No portability here!

Building an image

```
singularity build lolcow.simg shub://GodloveD/lolcow
```



Recipe file

```
BootStrap: debootstrap
OSVersion: stable
MirrorURL: http://ftp.us.debian.org/debian/

%runscript
    echo "This is what happens when you run the container..."

%post
    echo "Hello from inside the container"
    apt-get update
    apt-get -y install fortune cowsay lolcat
    apt-get clean
```

- Other keywords:
 - ➔ files, test, app

Recipe file

```
BootStrap: debootstrap
OSVersion: stable
MirrorURL: http://ftp.us.debian.org/debian/
```

Based on

```
%runscript
    echo "This is what happens when you run the container..."

%post
    echo "Hello from inside the container"
    apt-get update
    apt-get -y install fortune cowsay lolcat
    apt-get clean
```

- Other keywords:
 - ➔ files, test, app

Recipe file

```
BootStrap: debootstrap  
OSVersion: stable  
MirrorURL: http://ftp.us.debian.org/debian/
```

Based on

```
%runscript  
    echo "This is what happens when you run the container..."
```

What to do

```
%post  
    echo "Hello from inside the container"  
    apt-get update  
    apt-get -y install fortune cowsay lolcat  
    apt-get clean
```

- Other keywords:
 - ➔ files, test, app

Recipe file

```
BootStrap: debootstrap  
OSVersion: stable  
MirrorURL: http://ftp.us.debian.org/debian/
```

Based on

```
%runscript  
echo "This is what happens when you run the container..."
```

What to do

```
%post  
echo "Hello from inside the container"  
apt-get update  
apt-get -y install fortune cowsay lolcat  
apt-get clean
```

How to install

- Other keywords:
 - ➔ files, test, app

Recipe file

```
BootStrap: debootstrap  
OSVersion: stable  
MirrorURL: http://ftp.us.debian.org/debian/
```

Based on

```
%runscript  
echo "This is what happens when you run the container..."
```

What to do

```
%post  
echo "Hello from inside the container"  
apt-get update  
apt-get -y install fortune cowsay lolcat  
apt-get clean
```

How to install

```
[vagrant@localhost singularity]$ sudo singularity build test.simg centos.def
```

- Other keywords:
 - ➔ files, test, app

Run image

- You can run the “main” script

```
[omatt@lm3-m001 mpisingularity]$ singularity run ./mpi-user.simg  
This is what happens when you run the container...
```


- But also you can run any executables of the container


```
[omatt@lm3-m001 mpisingularity]$ singularity exec ./mpi-user.simg whoami  
omatt
```

Share with the community: Singularity Hub




Authorize Singularity Hub 2.0

**Singularity Hub 2.0** by [vsoch](#)
wants to access your [oliviermattelaer](#) account

**Personal user data**


Email addresses (read-only)

▼

**Repository webhooks and services**


Admin access

▼

**Organization webhooks**


Admin access

▼

**Commit statuses**

Read and write access

▼


**Organizations and teams**


Read-only access


▼

Authorize vsoch

Authorizing will redirect to
<https://www.singularity-hub.org>

 Not owned or
operated by GitHub

 Created
2 years ago

 More than 1K
GitHub users

- Link a git repository
- A recipe file
 - ➔ Singularity hub will build it
- Singularity build can automatically load containers from the hub

Singularity on the cloud (new)

Sylabs.ioSingularity LibraryRemote BuilderKeystore

Help ▾syabstest1 ▾

Build a Recipe File

Please attach build recipe by dragging & dropping, pasting from the clipboard or

selecting them

BootStrap: docker
From: ubuntu:latest

%runscript
 echo "This is what happens when you run the container..."

%post
 echo "Hello from inside the container"
 echo "Install additional software here"

✓ Build Recipe file is valid

Optional: Specify a location for the built image

Search Containers

Build

Using the Remote Builder from the Singularity CLI

To use the Remote Build Service, first make sure you have a Sylabs cloud token - [get one here](#). Save it to `~/.singularity/sylabs-token`, and then build using the `--remote` flag:

`singularity build --remote output.sif examples/docker/Singularity`

My Builds

Build ▾	Build Recipe	Submit Time ▾	Started Time ▾	Duration ▾	Library	Status ▾
docker:busybox:latest 756.00 KB	Build Recipe	2019-01-19T00:27:14+09:00	2019-01-19T00:29:23+09:00	06s	syabstest1/default/testproject2019018152657	Completed <div></div>

- Singularity is available on **Lemaitre3**
- We have the latest stable version (3.1.1)
- Part of the CECI formation
- We offer support for MPI (new)



MPI (new)

- MPI support requires
 - ➔ That you install the same slurm version as the one on our cluster
 - ➔ That you have the same version of mpi on the machine



- So you need **matching** pieces
 - ✓ We provide a starting container
 - ➔ Correct version of slurm
 - ➔ For each openmpi version
- You can use such container as base for your work

Conclusion

- Singularity
 - ➔ Nice way to share code with colleague
 - ➔ Portability and reproducibility
- Few commands to learn
 - ➔ Not that complicated!
- Need to be root on (one) machine
 - ➔ Ok that's annoying...
 - ➔ But you can use the singularity cloud
- MPI support
 - ➔ Dedicated image