Lambda/Container Dependencies

Tyler Caraza-Harter

Outline

General Dependencies (Slacker)

PyPI Dependencies (SOCK)

OpenLambda: Package Puller

Slacker

Background

- internship work done for Tintri (FAST paper, U.S. Patent 10,430,378)
- Docker used AUFS (another union FS) by default at the time
- overlayfs is similar (current default)

Slacker: Fast Distribution with Lazy Docker Containers

Tyler Harter, *University of Wisconsin—Madison*; Brandon Salmon and Rose Liu, *Tintri;*
Andrea C. Arpaci-Dusseau and Remzi H. Arpaci-Dusseau, *University of Wisconsin—Madison*

https://www.usenix.org/conference/fast16/technical-sessions/presentation/harter

https://www.usenix.org/system/files/conference/fast16/fast16-papers-harter.pdf

Images Analyzed

Language

clojure
gcc
golang
haskell
hylang
java
jruby
julia
mono
perl
php
pypy
python
r-base
rakudo-star
ruby
thrift

Linux Distro

alpine
busybox
centos
cirros
crux
debian
fedora
mageia
opensuse
oraclelinux
ubuntu
ubuntu-debootstrap
ubuntu-upstart

Database

cassandra
crate
elasticsearch
mariadb
mongo
mysql
percona
postgres
redis
rethinkdb

Web Framework

django
iojs
node
rails

Web Server

glassfish
httpd
jetty
nginx
php-zendserver
tomcat

Other

drupal
ghost
hello-world
jenkins
rabbitmq
registry
sonarqube

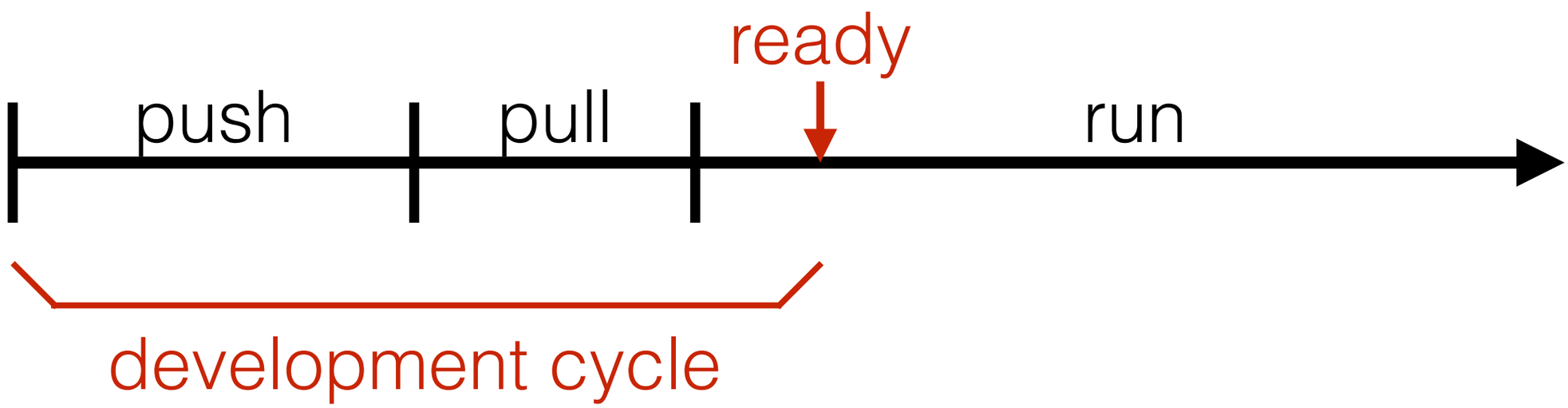
HelloBench

Goal: stress container startup

- including push/pull
- **57 container images** from Docker HUB
- run simple “hello world”-like task
- wait until it’s done/ready

Development cycle

- distributed programming/testing



HelloBench

Goal: stress container startup

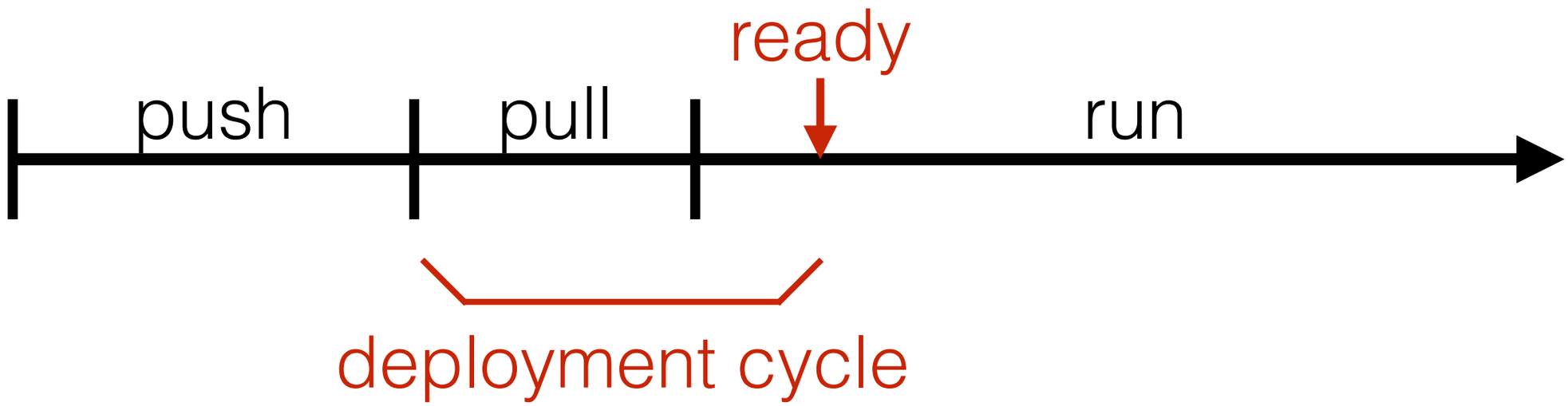
- including push/pull
- **57 container images** from Docker HUB
- run simple “hello world”-like task
- wait until it’s done/ready

Development cycle

- distributed programming/testing

Deployment cycle

- flash crowds, rebalance

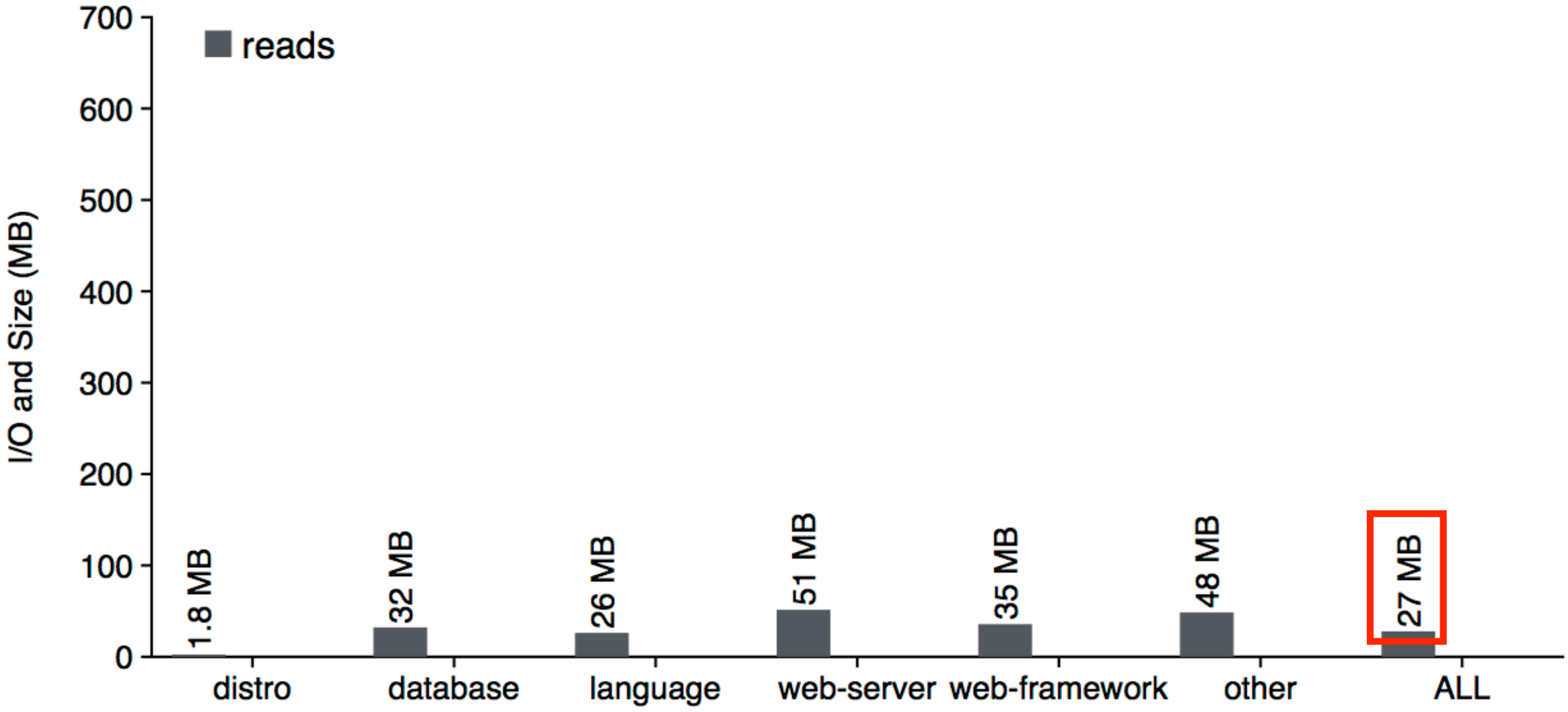


HelloBench questions

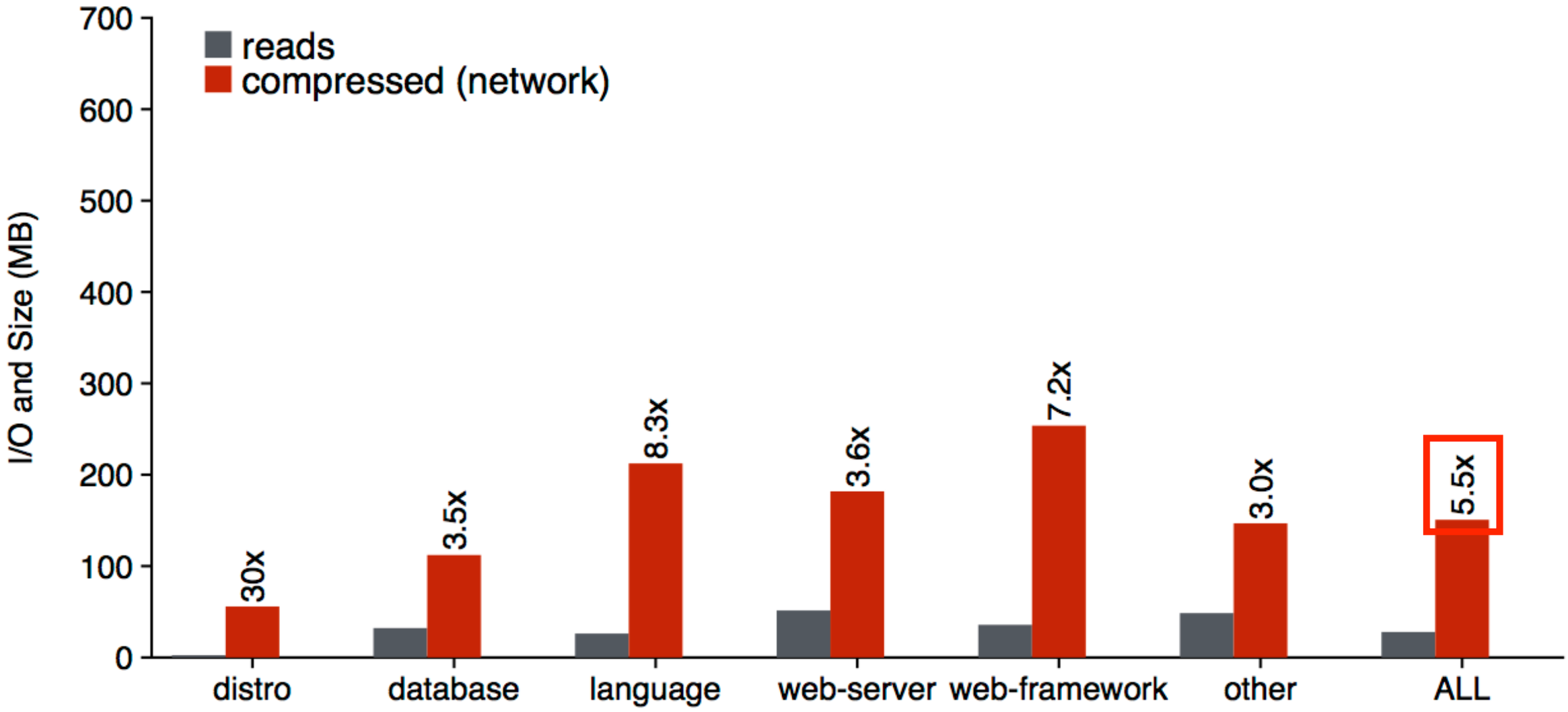
How much image data is needed for container startup?

How similar are reads between runs?

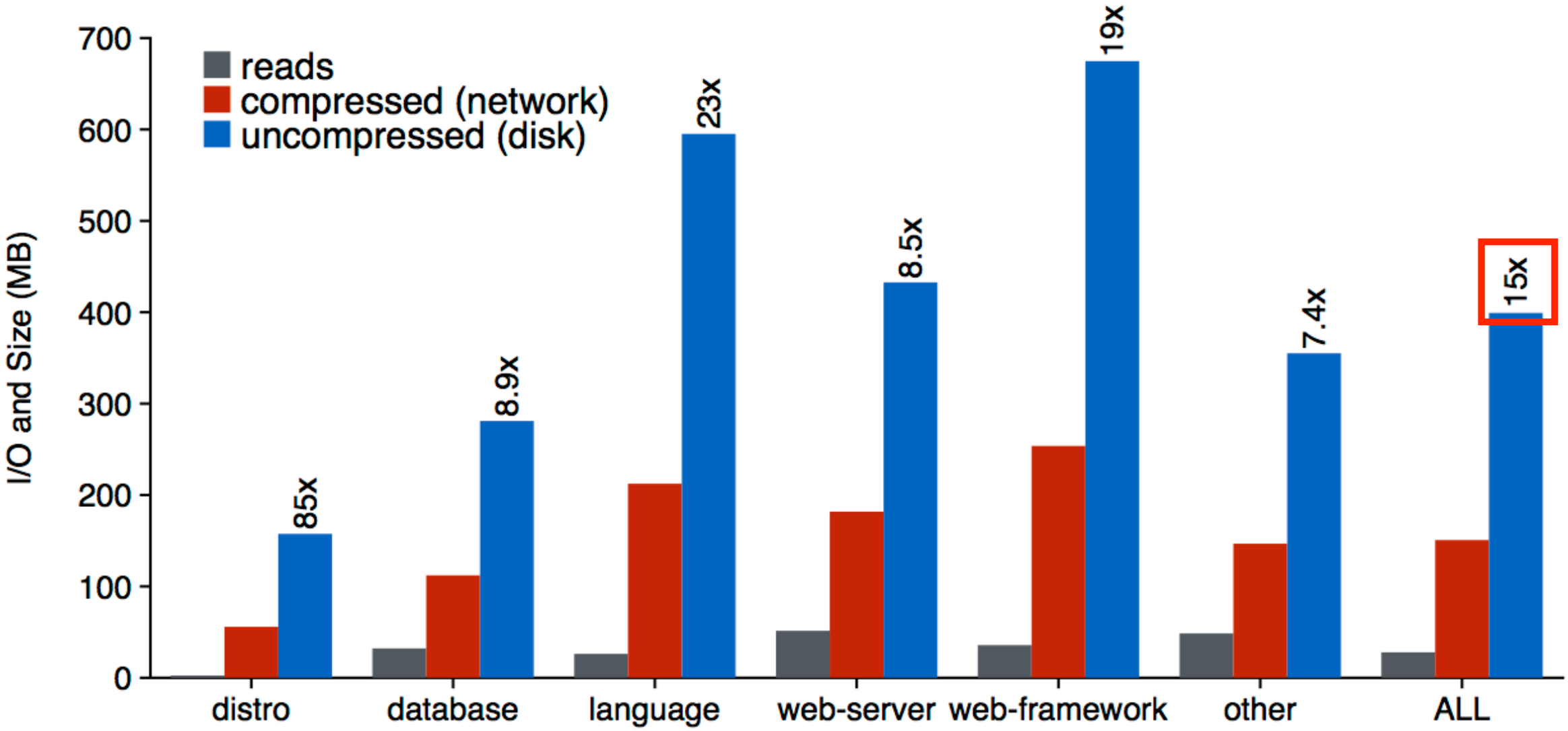
Container amplification



Container amplification



Container amplification



only 6.4% of data needed during startup

HelloBench questions

How much image data is needed for container startup?

- 6.4% of data is needed
- **design implication**: lazily fetch data

How similar are reads between runs?

HelloBench questions

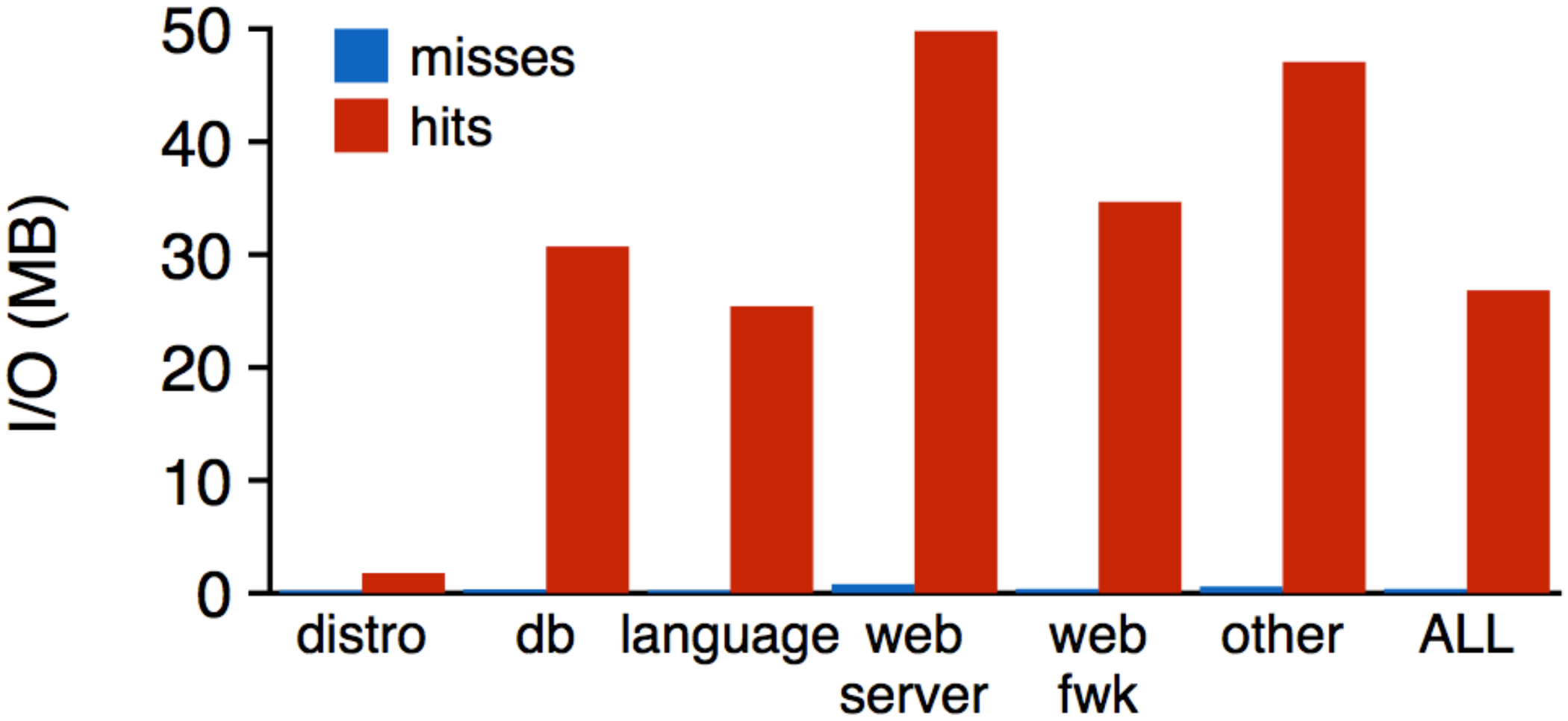
How much image data is needed for container startup?

- 6.4% of data is needed
- **design implication**: lazily fetch data

How similar are reads between runs?

Repeat runs

measure hits/misses for second of two runs



up to 99% of reads could be serviced by a cache

HelloBench questions

How much image data is needed for container startup?

- 6.4% of data is needed
- **design implication**: lazily fetch data

How similar are reads between runs?

- containers from same image have similar read patterns
- **design implication**: share cache state between containers

Speculate: these are for dependencies in general -- do we expect these patterns to whole for PyPI?

Outline

General Dependencies (Slacker)

PyPI Dependencies (SOCK)

OpenLambda: Package Puller

SOCK Analysis of Python Packages

Analysis part

- created PyPI mirror, analyzed all versions of 101K packages in 2018 (PyPI now has 461K projects)
- each package might have multiple modules, runnable programs, etc
- how long does it take to download/install/import?
- to what extent are Packages self contained?

SOCK: Rapid Task Provisioning with Serverless-Optimized Containers

Edward Oakes, Leon Yang, Dennis Zhou, and Kevin Houck, *University of Wisconsin-Madison;* Tyler Harter, *Microsoft, GSL;* Andrea C. Arpaci-Dusseau and Remzi H. Arpaci-Dusseau, *University of Wisconsin-Madison*

https://www.usenix.org/system/files/conference/atc18/atc18-oakes.pdf

Sizes

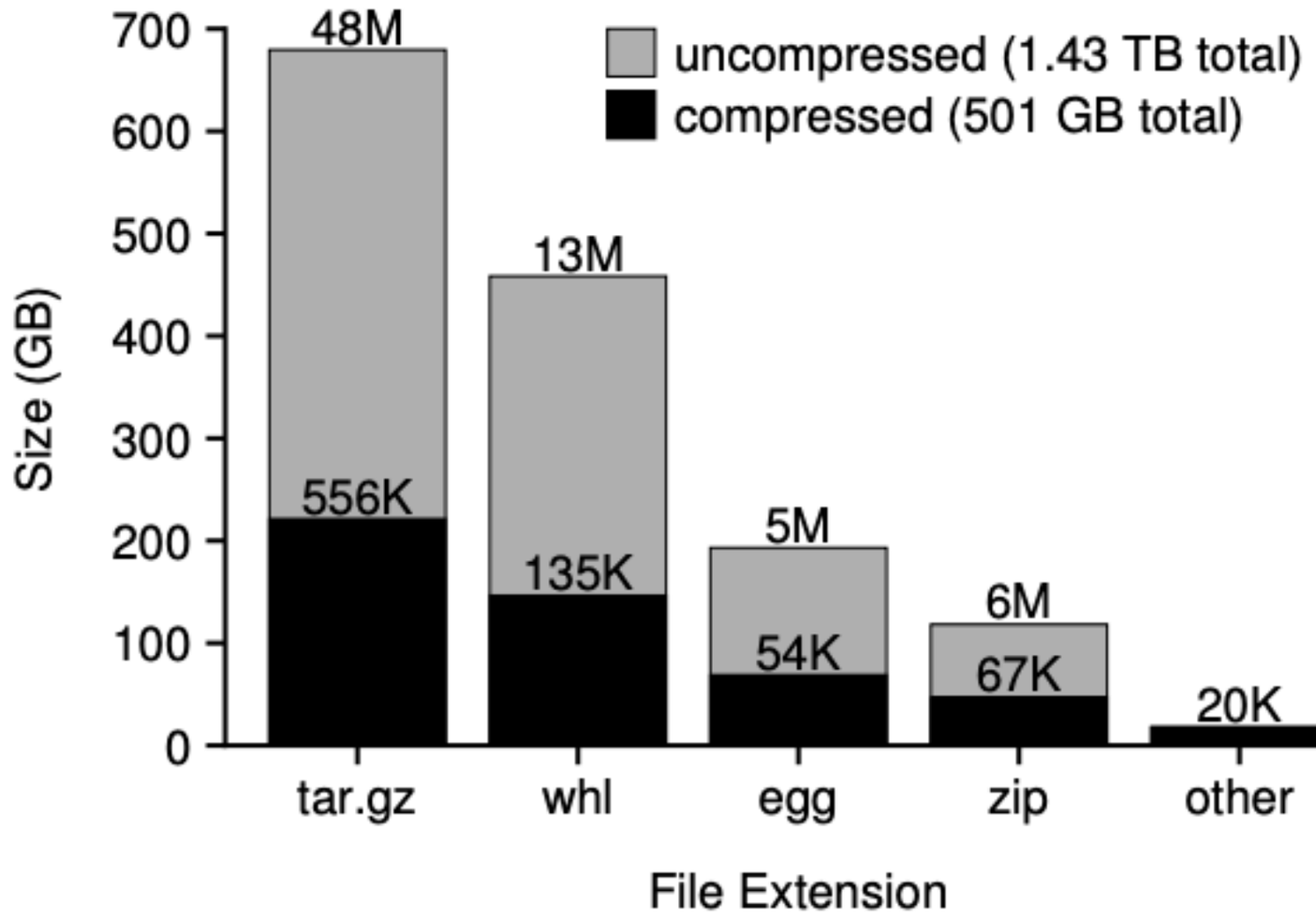


Figure 8. PyPI Package Data. *The size of the PyPI repository is shown, compressed and uncompressed, by file type (as of Mar 31, 2017). Bar labels show file counts.*

a single machine can host much (or all) of the packages

File Overlap

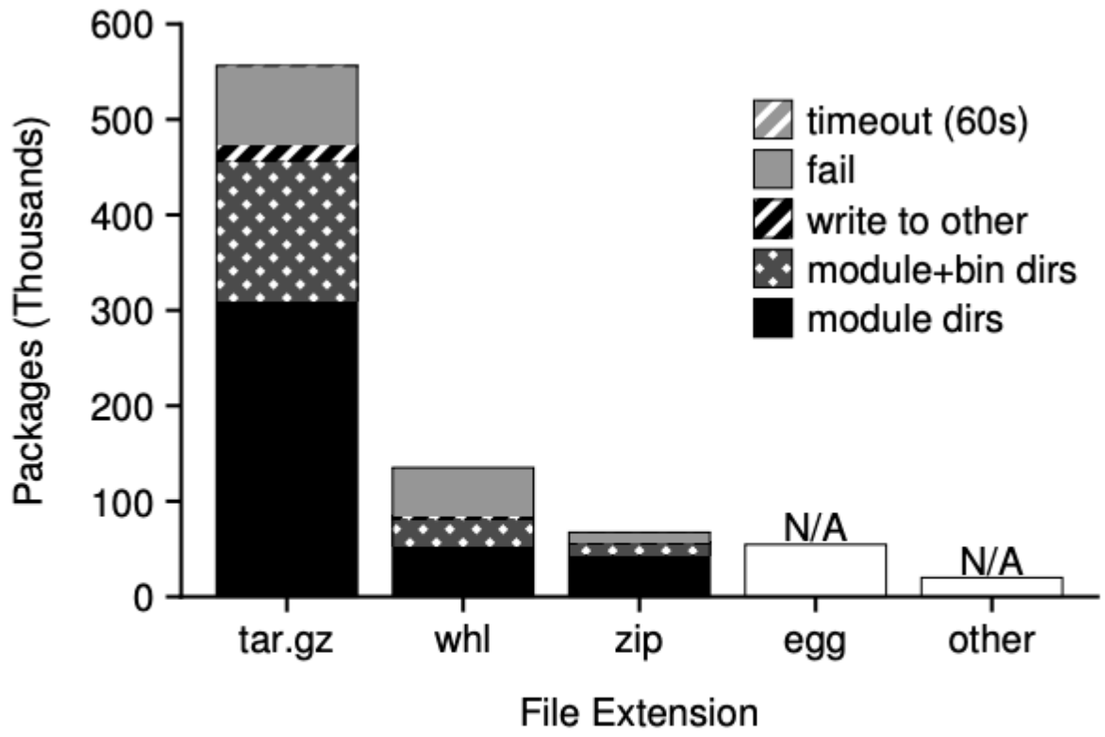


Figure 9. File-System Modifications. *The bars breakdown installations by the types of writes to the file system. The egg and other files can be used without extraction.*

Implication: these Dockerfiles should USUALLY have the same result

Dockerfile

```
RUN pip install pkgA
RUN pip install pkgB
```

Dockerfile

```
RUN pip install pkgB
RUN pip install pkgA
```

unnecessary strict ordering prevents **parallelism**, and worse prevents **general caching**

Startup Costs

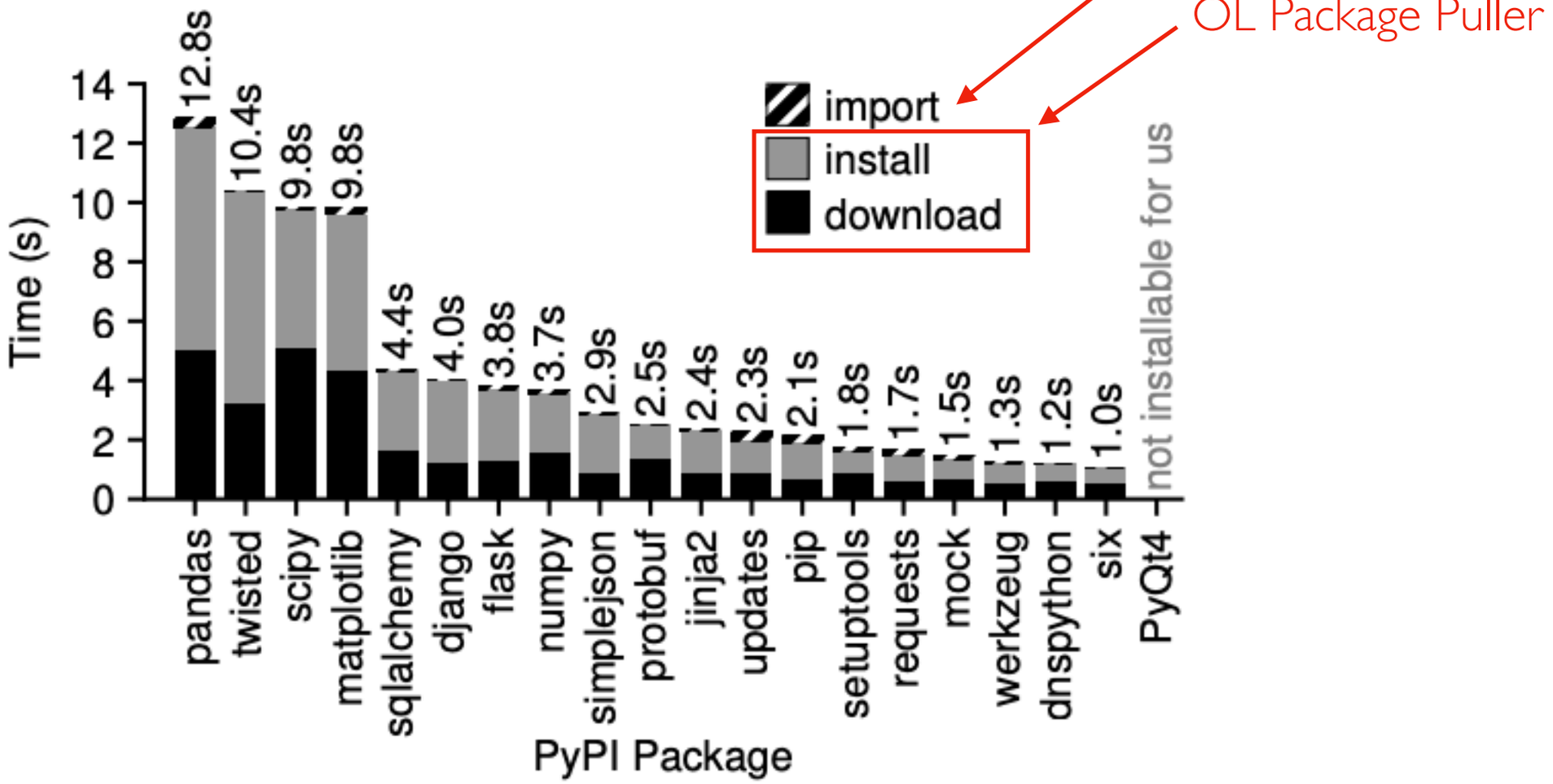


Figure 7. Startup Costs. *The download, install, and import times are shown for 20 popular Python packages, ordered by total initialization time.*

download+install dominate, but "import" matters for low-latency serverless platforms

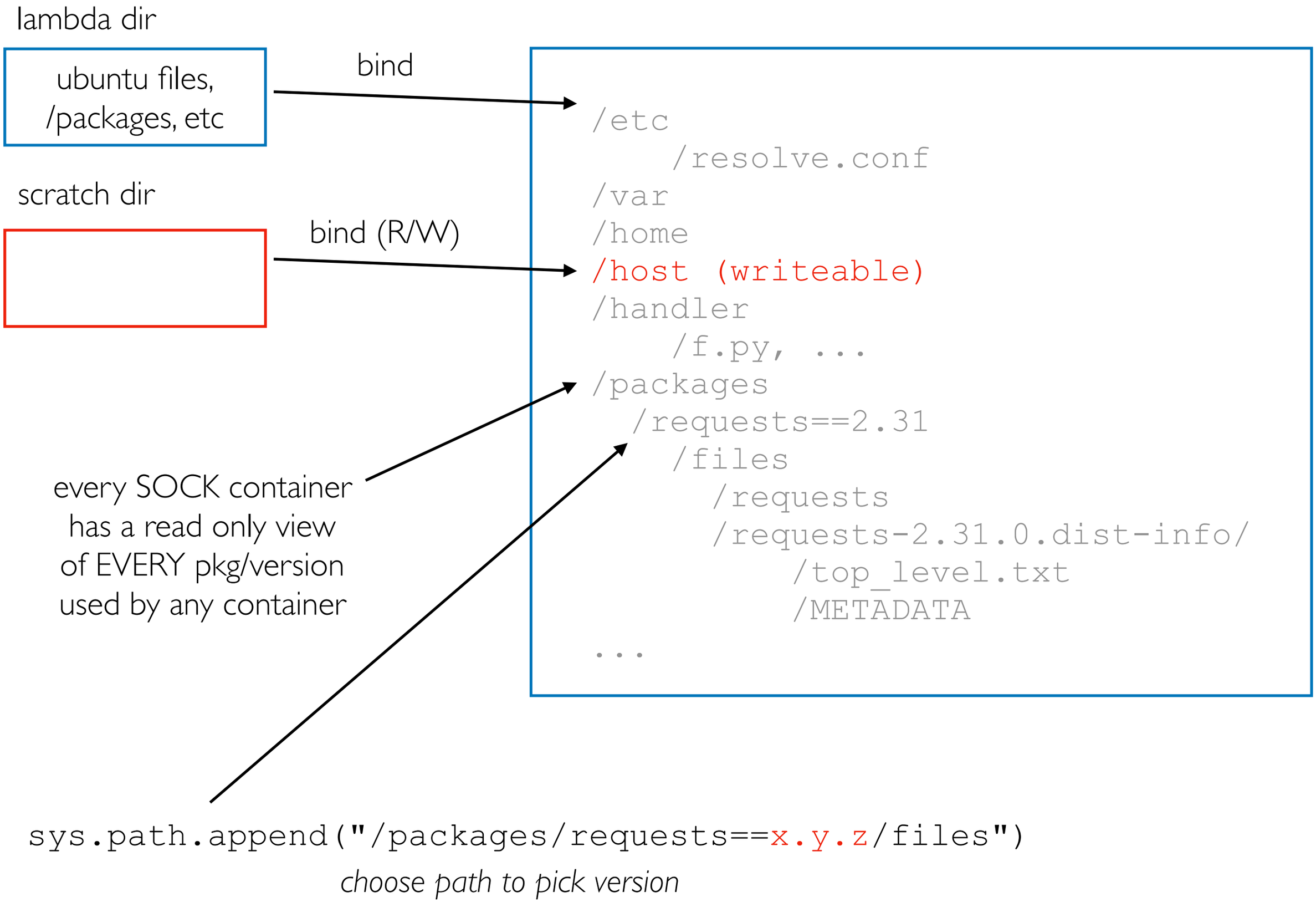
Outline

General Dependencies (Slacker)

PyPI Dependencies (SOCK)

OpenLambda: Package Puller

Root FS of a Normal SOCK Container



Root FS of an Installer SOCK Container

lambda dir

ubuntu files,
/packages, etc

sub directory:
/packages/requests==**2.30**

```
/files
```

bind

bind
(R/W)

```
/etc
    /resolve.conf
/var
/home
/host (writeable)
    /files
```

install modules here

```
/handler
    /f.py, ...
/packages
  /requests==2.31
    /files
      /requests
      /requests-2.31.0.dist-info/
          /top_level.txt
          /METADATA
  /requests==2.30
    /files

...
```

Command (in container):

pip3 install --no-deps requests==2.30 \
--cache-dir /tmp/.cache -t /host/files

Root FS of an Installer SOCK Container

lambda dir

ubuntu files,
/packages, etc

sub directory:
/packages/requests==**2.30**

```
/files
    /requests
```

bind

bind
(R/W)

```
/etc
    /resolve.conf
/var
/home
/host (writeable)
    /files
        /requests
/handler
    /f.py, ...
/packages
  /requests==2.31
    /files
      /requests
      /requests-2.31.0.dist-info/
          /top_level.txt
          /METADATA
  /requests==2.30
    /files
        /requests
...
```

install modules here

Command (in container):

pip3 install --no-deps requests==2.30 \
--cache-dir /tmp/.cache -t /host/files

Root FS of an Installer SOCK Container

packages could have any number of top-level modules

often there is one module with the same name as the package, but this isn't required

```
Metadata-Version: 2.1
Name: requests
Version: 2.31.0
Summary: Python HTTP for Humans.
Home-page: https://requests.readthedocs.io
...
Requires-Python: >=3.7
Description-Content-Type: text/markdown
Requires-Dist: charset-normalizer (<4,>=2)
Requires-Dist: idna (<4,>=2.5)
Requires-Dist: urllib3 (<3,>=1.21.1)
Requires-Dist: certifi (>=2017.4.17)
Provides-Extra: security
Provides-Extra: socks
Requires-Dist: PySocks (!=1.5.7,>=1.5.6) ; extra == 'socks'
Provides-Extra: use_chardet_on_py3
Requires-Dist: chardet (<6,>=3.0.2) ; extra == 'use_chardet_on_py3'

... Markdown documentation...
```

uh oh! if you only specify versions for direct dependencies, you might not get reproducibility due to range versions

```
/requests-2.31.0.dist-info/
    /top_level.txt
    /METADATA
/requests==2.30
  /files
    /requests
...
```

 # simple install
pip3 install requests

 # version and extras
pip3 install requests[socks]==2.30

```
# OpenLambda code for parsing Requires-Dist
os_name = os.name
sys_platform = sys.platform
platform_machine = platform.machine()
platform_python_implementation = platform.python_implementation()
...
extra = '' # TODO: support extras

def matches(markers):
    return eval(markers)
```

```
Meta
Name
Vers
Summ
Home
...
Requ
Desc
Requ
Requ
Requires-Dist: urllib3 (<3,>=1.21.1)
Requires-Dist: certifi (>=2017.4.17)
Provides-Extra: security
Provides-Extra: socks
Requires-Dist: PySocks (!=1.5.7,>=1.5.6) ; extra == 'socks'
Provides-Extra: use_chardet_on_py3
Requires-Dist: chardet (<6,>=3.0.2) ; extra == 'use_chardet_on_py3'

... Markdown documentation...
```

Tapping into pip's logic was too hard (at least at the time). OL implements a subset of this from scratch: https://peps.python.org/pep-0508

- extras: not supported
- range versions: not supported
- environment markers: supported

Recursive Install

```
Requires-Python: >=3.7
Description-Content-Type: text/markdown
Requires-Dist: charset-normalizer (<4,>=2)
Requires-Dist: idna (<4,>=2.5)
Requires-Dist: urllib3 (<3,>=1.21.1)
Requires-Dist: certifi (>=2017.4.17)
Provides-Extra: security
Provides-Extra: socks
Requires-Dist: PySocks (!=1.5.7,>=1.5.6) ; extra == 'socks'
```

Recursive install

- requests=x.y.z will be installed first (in a sandbox)
- then charset-normalizer, idna, urllib3, and certify (each in its own sandbox)
- then any dependencies of those (own sandbox)

Note on versioning, these are considered different, even if current is 2.30

- pip3 install requests
- pip3 install requests==2.30

Install Resources

```
{
  "worker_dir": "/root/workshop/tyler/tyler-ol/worker",
  "worker_url": "localhost",
  "worker_port": "5005",
  "log_output": true,
  "sandbox": "sock",
  "server_mode": "lambda",
  "registry": "/root/workshop/tyler/tyler-ol/registry",
  "registry_cache_ms": 5000,
  "Pkgs_dir": "/root/workshop/tyler/tyler-ol/lambda/packages",
  "pip_mirror": "",
  "mem_pool_mb": 30000,
  "sock_base_path": "/root/workshop/tyler/tyler-ol/lambda",
  "sandbox_config": {},
  "docker_runtime": "",
  "limits": {
    "procs": 80,
    "mem_mb": 50,
    "cpu_percent": 100,
    "max_runtime_default": 30,
    "swappiness": 0,
    "installer_mem_mb": 500
  }
}
```

install is resource intense (e.g., need to compile C code), so installer Sandboxes can have a different memory limit.