The Popper Container-native Workflow Engine

Ivo Jimenez

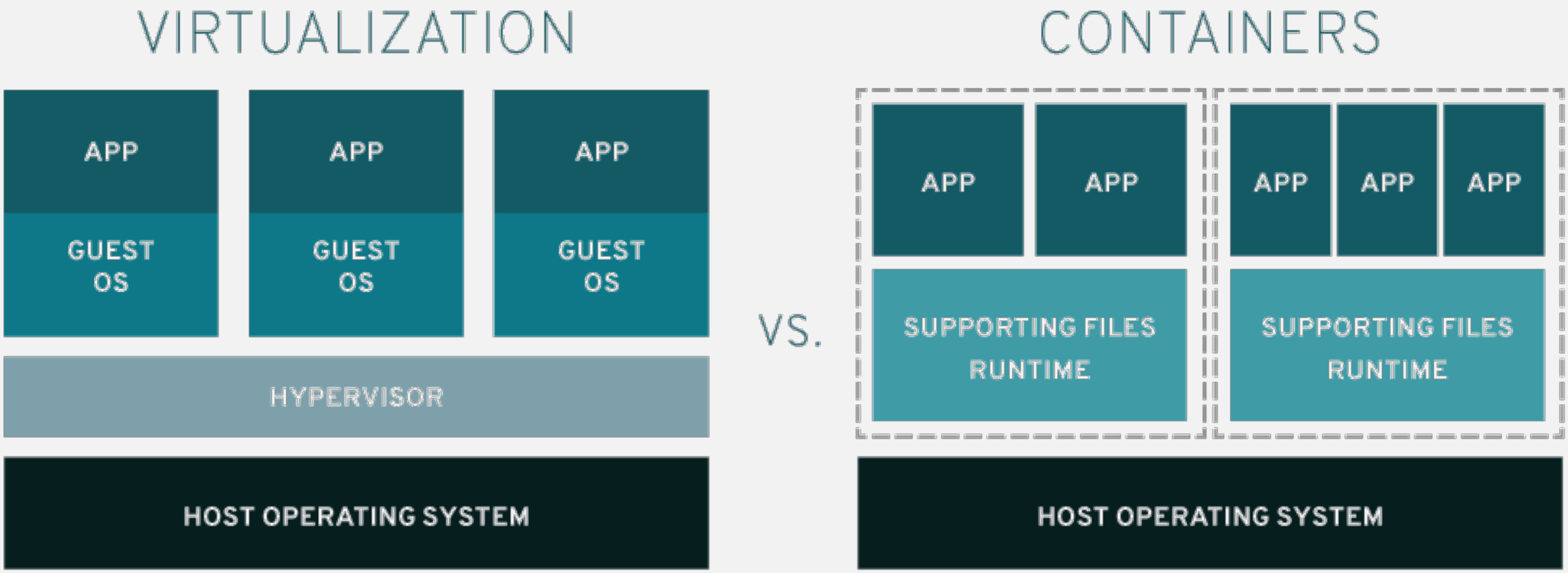
<ivotron.me>

Research Scientist and CROSS Incubator Fellow

UC Santa Cruz

What is a container?

What is a container?



Main benefit of using containers

Bring Your Own Environment (BYOE) to shared infrastructure

Main benefit of using containers

Bring Your Own Environment (BYOE) to shared infrastructure

```
docker run mattrayner/lamp:latest-1804
```

Main benefit of using containers

Bring Your Own Environment (BYOE) to shared infrastructure

```
docker run tensorflow/tensorflow:2.1.1-gpu-jupyter
```

What is the container-native paradigm?

What is the container-native paradigm?

Use containers for everything:

- Build software, pre-process data, deploy software, allocate compute resources, run tests, analyze data, validate results, generate manuscripts, etc.

What is the container-native paradigm?

Use containers for everything:

- Build software, pre-process data, deploy software, allocate compute resources, run tests, analyze data, validate results, generate manuscripts, etc.

Software doesn't get installed directly on a host machine; it is running in containers instead*

*other than personal productivity tools such as a text editor, web browser, email reader, calendar app, etc.

Practical problems that arise when working under the container-native paradigm

Practical problems that arise when working under the container-native paradigm

- Dealing with multi-container workflows
    - Lack of out-of-the-box support for complex application testing and prototyping

Practical problems that arise when working under the container-native paradigm

- Dealing with multi-container workflows
    - Lack of out-of-the-box support for complex application testing and prototyping
- Myriad of container runtimes and engines
    - Docker, Podman, LXD, Singularity, Charliecloud, …

Practical problems that arise when working under the container-native paradigm

- Dealing with multi-container workflows
    - Lack of out-of-the-box support for complex application testing and prototyping
- Myriad of container runtimes and engines
    - Docker, Podman, LXD, Singularity, Charliecloud, …
- Lack of common orchestration platform support
    - SLURM, Kubernetes, CI services, …

Practical problems that arise when working under the container-native paradigm

- Dealing with multi-container workflows
    - Lack of out-of-the-box support for complex application testing and prototyping
- Myriad of container runtimes and engines
    - Docker, Podman, LXD, Singularity, Charliecloud,
- Lack of common orchestration platform support
    - SLURM, Kubernetes, CI services, …

Popper

Practical problems that arise when working under the container-native paradigm

- Dealing with multi-container workflows
  - Lack of out-of-the-box support for complex application testing and prototyping
- Myriad of container runtimes and engines
  - Docker, Podman, LXD, Singularity, Charliecloud, …
- Lack of common orchestration platform support
  - SLURM, Kubernetes, CI services, …

Popper


```
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

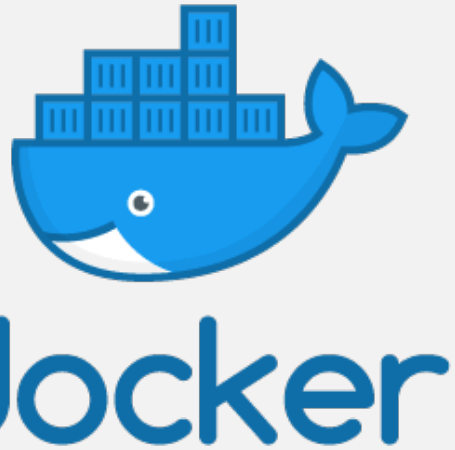
https://github.com/getpopper/popper

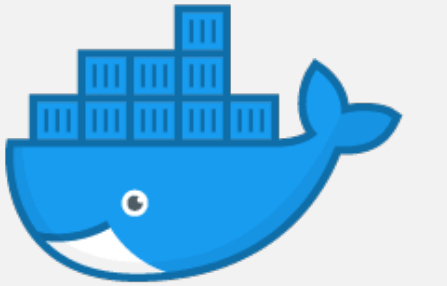

```
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

https://github.com/getpopper/popper

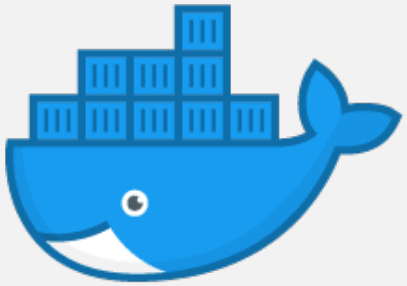
Practical problems that arise when working under the container-native paradigm

- Dealing with multi-container workflows
    - Complex application testing and prototyping becomes difficult to reproduce if done by hand
- Myriad of container runtimes and engines
    - Docker, Podman, LXD, Singularity, Charliecloud, …
- Lack of common orchestration platform support
    - SLURM, Kubernetes, CI services, …


docker

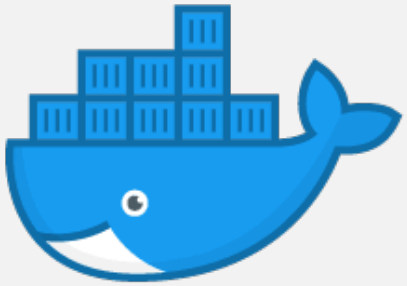


docker



docker

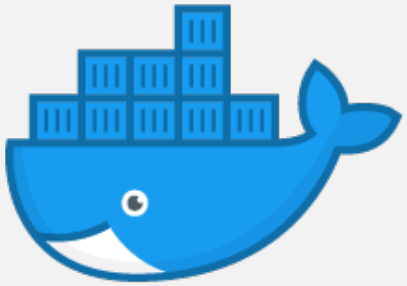
podman



docker

podman

lxd

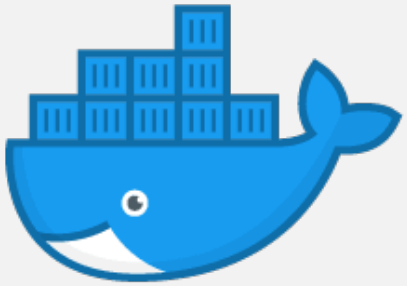


docker

podman

Charliecloud

lxd

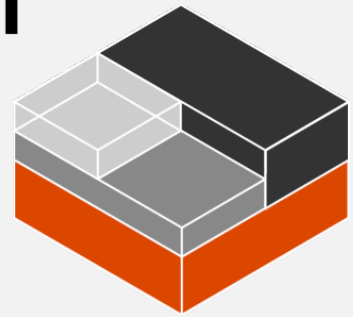


docker

kata
containers

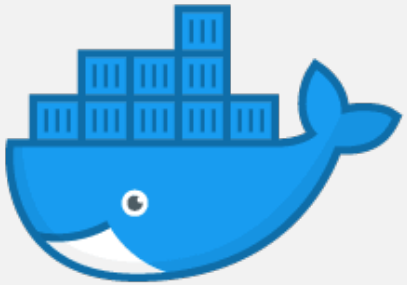
podman

Charliecloud



lxd

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```



docker

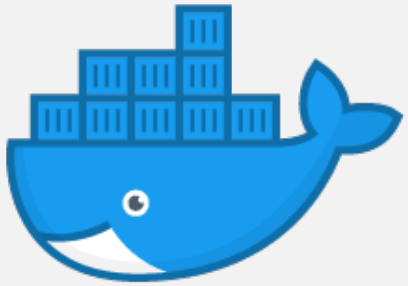
kata containers

podman

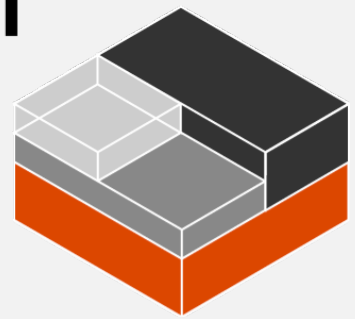
Charliecloud

lxd

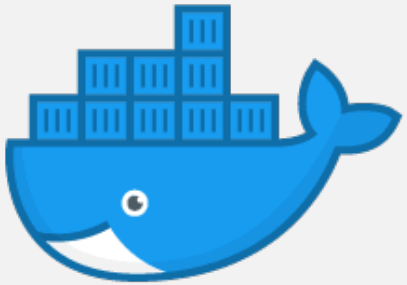
```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```



podman



```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```



docker

kata containers

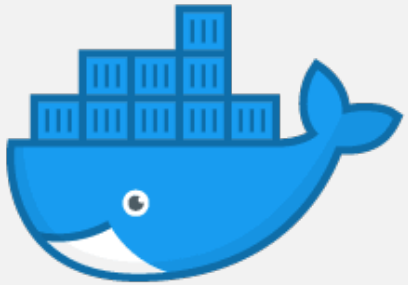
podman

Charliecloud

lxd

Popper

```
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

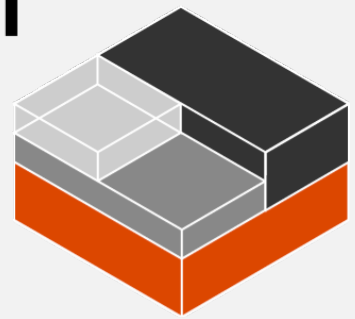


docker

kata containers

podman

Charliecloud

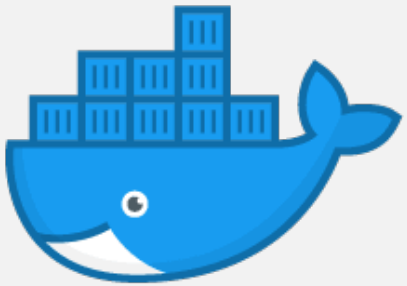


lxd

Popper

```
$> popper run
```

```
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

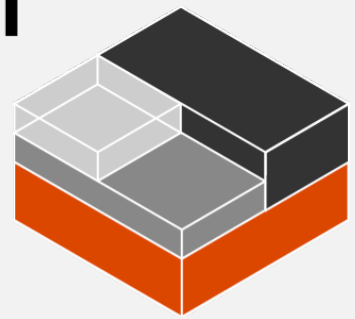


docker

kata containers

podman

Charliecloud



lxd

Popper

```
$> popper run --engine podman
```

```
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

docker

kata containers

podman

lxd

Charliecloud

Practical problems that arise when working under the container-native paradigm

- Dealing with multi-container workflows
    - Complex application testing and prototyping becomes difficult to reproduce if done by hand
- Myriad of container runtimes and engines
    - Docker, Podman, LXD, Singularity, Charliecloud, …
- Lack of common orchestration platform support
    - SLURM, Kubernetes, CI services, …

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

kubernetes

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

HTCondor

kubernetes

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

HTCondor

kubernetes

slurm
workload manager

GRID ENGINE

Popper

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

HTCondor

kubernetes

slurm
workload manager

GRID ENGINE

Popper

```
$> popper run
```

```
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

HTCondor

kubernetes

slurm
workload manager

GRID ENGINE

Popper

```
$> popper run --resman slurm
```

```
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

HTCondor

kubernetes

slurm
workload manager

GRID ENGINE

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

Buildkite

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

Buildkite

```yaml
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

Buildkite

circleci

shippable

Popper

```
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

Buildkite

circleci

shippable

Popper

```
$> popper ci --service travis
```

```
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

- id: delete existing jobs
  uses: popperized/bin/sh@master
  args: [rm, -fr, sweep/jobs]

- id: install sweepj2
  uses: popperized/python-actions@master
  args: [pip, install, sweepj2]

- id: generate sweep
  uses: jefftriplett/python-actions@master
  args: [
    "sweepj2",
    "--template", "./sweep/script.j2",
    "--space", "./sweep/space.yml",
    "--output", "./sweep/jobs/",
    "--make-executable"
  ]

- id: run sweep
  uses: popperized/spack@master
  args: [run-parts, ./sweep/jobs]
```

Buildkite

circleci

shippable

One workflow to rule them all

Example workflows

- Ceph benchmarking: deploy K8S on baremetal, Ceph via Rook; run benchmarks, plot results on Jupyter notebooks
- C++ project: package dev environment in container; build and run unit tests; prepare and run non-functional tests
- Machine learning: build C++ library, install python packages, download datasets, train and evaluate models, show results.
- Others: genomics, computational science, geosciences, etc.

One workflow to rule them all

`github.com/getpopper/popper`

ivotron.me

getpopper / popper

Watch 13 | Star 178 | Fork 45

Code | Issues 31 | Pull requests 5 | Actions | Security | Insights

Branch: master | Go to file | Clone

fsr313 committed 1d7daf2 3 days ago ... ✓ — 582 commits | 3 branches | 21 tags

docs	Revert "Add paramiko as dependency to allow DOCKER_...	3 days ago
examples	Add ansible workflow example	12 days ago
src	add sanitization for image and container ids (#867)	3 days ago
.codecov.yml	updates codecov configuration	2 months ago
.gitignore	Move extras/ to scripts/	20 days ago
.pep8speaks.yml	adds black to travis and reformats entire codebase (#8...	2 months ago
.popper.yml	Add --allow-undefined-secrets-in-ci popper run flag	4 days ago
.travis.yml	Add --allow-undefined-secrets-in-ci popper run flag	4 days ago
CODE_OF_CONDUCT.md	Update README.md, add CONTRIBUTING.md, and COD...	3 years ago
CONTRIBUTING.md	Fix link in CONTRIBUTING.md	12 days ago
LICENSE	Create LICENSE	3 years ago
README.md	Add link to pre-print pdf and fix bibtex link	11 days ago
install.sh	map env vars into the popper container from current sh...	15 days ago

README.md

Popper

Popper is a tool for defining and executing container-native workflows in Docker, as well as other container engines. With Popper, you define a workflow in a YAML file, and then execute it with a single command. A workflow file looks like this:

```
steps:
# download CSV file with data on global CO2 emissions
- id: download
  uses: docker://byrnedo/alpine-curl:0.1.8
  args: [-LO, https://github.com/datasets/co2-fossil-global/raw/master/global.csv]
```

About

Container-native workflow execution engine.

reproducibility | cli | devops | sciops | containers | workflows | docker | singularity | podman | workflow-engine

Readme

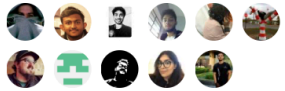
MIT License

Releases 21

v2.6.0 Latest
22 days ago

+ 20 releases

Contributors 29



+ 18 contributors

Languages

Python 98.4% | Shell 1.5% | Dockerfile 0.1%

OSS Research Experience Project Ideas

- Support more container engines
  - podman, charliecloud, lxd
- Support other resource managers
  - Kubernetes, HTCondor, GridEngine
- Reproducible performance tests
  - Ceph, SkyhookDM, SPDK, DPDK, Seastar, etc.
- Reproducible workflows in other domains:
  - Computational research, machine learning, etc.

10+ Years of Mentoring Experience

UNIVERSITY OF CALIFORNIA
SANTA CRUZ

Google Summer of Code

mozilla
FOUNDATION

CROSS | CENTER FOR RESEARCH IN OPEN SOURCE SOFTWARE

eLIFE

HP Labs

One workflow to rule them all

`github.com/getpopper/popper`

ivotron@ucsc.edu