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?

**VIRTUALIZATION**

- App
  - Guest OS

- App
  - Guest OS

- App
  - Guest OS

**CONTAINERS**

- App
  - Supporting files runtime

- App
  - Supporting files runtime

- App
  - Supporting files runtime

VS.

**HOST OPERATING SYSTEM**

**HYPERVERSOR**
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, …
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, …
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:* jeff triplett/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, …
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]
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpl]

- 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]
steps:

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

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

- 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: jeff triplett/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]
$>

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

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

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

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

- id: install sweepj2
  uses: poppered/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: poppered/spack@master
  args: [run-parts, ./sweep/jobs]
steps:
- id: install lulesh
  uses: popplerized/spack@master
  args: [spack, install, -j8, lulesh+mpi]

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

- id: install sweepj2
  uses: popplerized/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: popplerized/spack@master
  args: [run-parts, ./sweep/jobs]
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpl]

- 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]
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, -rf, 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]
$> popper run --resman slurm

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

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

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

- 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]
steps:
  - id: install lulesh
    uses: poppered/spack@master
    args: [spack, install, -j8, lulesh+mpl]

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

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

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

  - id: run sweep
    uses: poppered/spack@master
    args: [run-parts, ./sweep/jobs]
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]
steps:
  - id: install lulesh
    uses: popperized/spack@master
    args: [spack, install, -j8, lulesh+mpl]
  - 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]
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]
steps:
- id: install lulesh
  uses: popperized/spack@master
  args: [spack, install, -j8, lulesh+mpl]

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

- 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]
$> popper ci --service travis

**Steps:**
- **id**: install lulesh
  - **uses**: popperized/spack@master
  - **args**: [spack, install, -j8, lulesh+mpl]

- **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]
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
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://byrdneo/alpine-curl@v1.0.1
```
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

CROSS CENTER FOR RESEARCH IN OPEN SOURCE SOFTWARE

eLIFE

HP Labs

UNIVERSIDAD DE SONORA 1942
One workflow to rule them all

github.com/getpopper/popper

ivotron@ucsc.edu