Container in HPC - 2022 Edition

FOSDEM’22 talk about what to look out for in 2022

Christian Kniep, 05.Feb.2022
I did talk about the runtime aspect in 2021.

ReCap: Do not fight over which runtime to use; they all do the same. Insist on OCI runtime and image spec!
Overview
Challenges for High Performance Containers

... through the lens of (distributed) AI/ML

- single node, total isolation
  - simple container
  - container volume

- single node, GPU, isolated storage
  - container volume
  - device passthrough

- single node, GPU, shared storage
  - container volume
  - shared file-system (POSIX)

- multi node, NIC/GPU, shared storage
  - MPI / SLURM
  - user/kernel driver

Complexity

Maturity

High Performance Computing
## Container Ecosystem Segments

From the bottom up...

<table>
<thead>
<tr>
<th>Who</th>
<th>What</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Runtime</td>
<td>Create a (containerized) process which had (at least) an isolated file-system view.</td>
<td>runc, crun, youki</td>
</tr>
<tr>
<td>Engine</td>
<td>Lifecycle of container images and containers on a single node. Create snapshot of image, setup network and generate config to be picked up by the runtime.</td>
<td>Containerd, Sarus, Podman, Singularity</td>
</tr>
<tr>
<td>Scheduler</td>
<td>Orchestrates container placement (usually across a cluster of multiple instances). Provides API to interact.</td>
<td>Docker (Swarm), Kubernetes, Slurm</td>
</tr>
<tr>
<td>Build Tools</td>
<td>Creates container images to encapsulate software stacks. Either based on deriving a Dockerfile to be build or create a snapshot of a filesystem.</td>
<td>Spack/EasyBuild, HPCM, Dockerfile, OpenHPC</td>
</tr>
<tr>
<td>Distribution</td>
<td>Tooling, APIs to distribute OCI container images.</td>
<td>DockerHub, Github Container Registry, ECR</td>
</tr>
</tbody>
</table>
Build/Distribute/Schedule
Things to Do/Not Do in 2022
Do **NOT** focus on Runtimes!

It’s a pre-COVID discussion!
# Some Runtimes

<table>
<thead>
<tr>
<th>Runtimes</th>
<th>Engines</th>
</tr>
</thead>
<tbody>
<tr>
<td>youki (rust)</td>
<td>Sarus</td>
</tr>
<tr>
<td>crun (C)</td>
<td>containerd</td>
</tr>
<tr>
<td>runc (GOLANG)</td>
<td>Podman</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>All-in-One</th>
</tr>
</thead>
<tbody>
<tr>
<td>Singularity</td>
</tr>
<tr>
<td>NROOT(?)</td>
</tr>
<tr>
<td>Shifter</td>
</tr>
</tbody>
</table>
Do **NOT** start your container journey on Kubernetes!

Unless you are ‘born’ in K8s. Otherwise use the environment you know and love...
If you know SLURM, do get started there.

Running an HPC runtime of your (admins) choosing. It’s easier than you think!
Do build your image with AUTOMATION!

Or at least do it reproducibly
Do build your image with **AUTOMATION**!

Or at least do it reproducibly
Do think about how to annotate your images and compute resources!

So that your runtime, scheduler or registry gets some clues about preferences, constraints.