SC20 Tutorial

Online Questions

Q: Copy this and paste your question
A: We will answer it here

Q: Copy this and paste your question
A: We will answer it here

Feedback

Please feel free to make any suggestions for improvements here…

SC20 AWS Instances

Please mark an instance below to use for the hands-on portions by filling in your last name.

Hostname	**Participant's Last Name**
tut001	Lindemann
tut002	Cossey
tut003	
tut004	
tut005	
tut006	
tut007	
tut008	
tut009	
tut010	
tut011	
tut012	

tut013	
tut014	
tut015	
tut016	
tut017	
tut018	
tut019	
tut020	
tut021	
tut022	
tut023	
tut024	
tut025	
tut026	
tut027	
tut028	
tut029	
tut030	
tut031	
tut032	
tut033	
tut034	
tut035	
tut036	
tut037	
tut038	
tut039	
tut040	
tut041	
tut042	
tut043	
tut044	
tut045	
tut046	

tut047	
tut048	
tut049	
tut050	
tut051	
tut052	
tut053	
tut054	
tut055	
tut056	
tut057	
tut058	
tut059	
tut060	
tut061	
tut062	
tut063	
tut064	
tut065	
tut066	
tut067	
tut068	
tut069	
tut070	
tut071	
tut072	
tut073	
tut074	
tut075	
tut076	
tut077	
tut078	
tut079	
tut080	

tut081	
tut082	
tut083	
tut084	
tut085	
tut086	
tut087	
tut088	
tut089	
tut090	
tut091	
tut092	
tut093	
tut094	
tut095	
tut096	
tut097	
tut098	
tut099	

Previous Sessions

ECP20 Tutorial Online Questions

Please add your question below are mark it in **bold**

Q: Ask your question...

We will answer it.

SC19 Tutorial Online Questions

Please add your question below are mark it in **bold**

Q: Why should I use Shifter vs Singularity, and vice versa? What uses cases they handle better than the other?

(Shane) So I'm a developer of Shifter so I have some biases.
Singularity is easier to install and works around the concept of an image as a file. Singularity is much more popular and has an active developer community.
Shifter has a more centralized deployment model. So the images are shared. Shifter's development is still active but is much more limited.

Q: Who should I contact to be included in hpc-containers Slack channel?
Use the invite link https://bit.ly/hpccslack

Q: Would it be possible to describe/demo/point_to_an_example where Docker is used to build a multi-stage container (e.g. to build executables using icc, impi), which is then run by Singularity? I heard it's possible, I don't know exactly how it would go.
(My goal is to avoid having to ship Intel libraries within the container, I can guarantee all those will be available on host system)

A: maybe this documentation can help
https://sylabs.io/guides/3.5/user-guide/definition_files.html#multi-stage-builds

Shane: I don't have an example with Singularity but I can point to what we do for Shifter.
https://docs.nersc.gov/programming/shifter/intel/

I think that could be adapted to work with Singularity. The base images aren't public because they have the compilers, but I can put you in touch with the person who made them. If you can get a NERSC account, you could use those images directly.

Q: Every time we want to run with a new image, we have to log out of the previous session, download the image and as for a node?

A: I think this question was for shifter. No, you can specify an image on the command-line too. For example,...

shifter --image=busybox bash

The main advantage of specifying the image as part of the batch submission is the Shifter workload manager plugin can pre-mount the image so startups can be faster (especially at scale).

Shane: This is correct. Sometimes you need to run multiple images as part of a single job. In that case, you would just use the --image option for each shifter run.

Q: Can we safely delete the script file after creating the docker image? Does the docker make a copy of the script file?

A:Yes and no, you can rm the dockerfile, but docker won't keep a copy of it
You can, read the OCI json file on how it was built, but is not as helpful and version control best practices to do that. Always keep your Dockerfiles in git, and version them

(Shane) Deleting it will not cause the image to break. It would just impact your ability to rebuild the image.

Q: for hello MPI docker image - need to tag it first before push, right?

A: No need for tag but for <mydockerlogin>/imagename
The tag is optional, it will tag it lastest if no tag

Q: /usr/sbin/shifter: /app/app.py: no such file or directory
Srun error nid00694: task existed with exit 127
???????

A (Andrew)r: This is just an example. If you look at the example, it's using ubuntu:14:04 image, but that doesn't have /app/app.py in that image. So, you can try to run /bin/echo or something. Or, you can use your own container image! Eg: myuser123/mpihello:latest and run /bin/mpihello

Is anyone reading this?

(Shane) sorry about that. I will fix the example. As Andrew said it was meant to be generic, but it should be consistent with the other examples.

Q: next question

Spack Section

NOTE: Spack tutorial Tomorrow (Monday) 8-5! (preview here)

Q: Was Nix considered for the Spack use case? (ref [1], [2])

A: (Shane) Sameer may need to answer this, but Spack is a major part of the ECP deployment model so I think that is part of the reason the example uses it.
A: (Andrew) With containers, you have control of your environment, so you could use Nix inside a container. This is true for any package management tool, generally. With DOE/ECP, we are primarily focused on using Spack at this time.
A: (Todd) Yes, see the original spack paper; it is cited there as a major influence (Nix is awesome!). The main differences between Nix and Spack (w.r.t. HPC use cases, at least) are:

- Spack packages are templated; Nix packages are not (as much). You can swap compilers, MPI implementations, BLAS implementations, build options, compiler flags, and (as of 0.13), specific microarch targets with command line options in Spack -- it's intended for HPC users who want to experiment with different versions of packages.
- Spack lets you easily leverage installed packages like MPI. See https://spack.readthedocs.io/en/latest/build_settings.html#external-packages. AFAIK Nix doesn't have (easy) ways to link to external (e.g., vendor-supplied) libraries.
- Nix's package recipe language is a proprietary functional language. Spack packages are written in Python. We chose Python in hopes of broad uptake in HPC.
- Spack has spack.yaml/spack.lock files -- I do not believe Nix supports this yet (until Nix flakes). We think this is a really good model for reproducibility in containers (you get portable, "requirements-based" reproducibility with spack.yaml; you get "exact" reproducibility with spack.lock)
- Spack hashes are metadata hashes (they represent a configuration in the combinatorial configuration space). Nix hashes are hashes of the installed binary, so it's a bitwise level of reproducibility. We wanted something in between for Spack (so you can build the same environment for ARM, Power, skylake, etc.)
- Email/msg for more details (tgamblin@llnl.gov).

Q: Spack: when a new package is released, what is the lead time for incorporation into the Spack specs for installation?

A: (Shane) Probably best for Sameer, but I think it depends on the package and how much attention it gets. Anyone can create a PR to Spack for an update. So if there are people in the community that are staying on top of it, it could be very quick.

A: (Todd):

- For releases (https://github.com/spack/spack/releases) we (Spack) try to keep the package versions stable, so we wouldn't put new major versions there. But we are trying

to do releases much more frequently with help from the builds the E4S folks are doing -- these help us to test the packages and ensure they're robust.
- For the rolling develop branch, you could get a new version in in a day if you nag us on GitHub :). We cut new releases from this branch. So it can be very quick.
- You can also easily maintain a local Spack repo with your changes to packages that aren't upstreamed yet (see below).

Q: How are supported software stacks defined? What is the requirement to have a novell software package build and incorporated into Spack? I.e. I have novell package 'programX'. What has to happen for this to be available as a Spack delivered package?

A: I think you can define your own spack repositories. So they don't have to go into a specific centralized one.

A: (Todd):
- We have a number of ways of defining software stacks. See:
  - Spack environments: Docs, Tutorial
  - Spack stacks (combinatorial environments): Docs, Tutorial
  - These are ways to store configuration for your own builds of spack packages -- packages are the build recipes but they're intended to work out of the box -- you likely not need to tweak them.
- If you want to maintain your own Spack packages separately from the mainline (e.g. for proprietary or export controlled codes), you can do that with *repositories:*
  - https://spack.readthedocs.io/en/latest/repositories.html
  - Packages in repositories can extend or replace builtin packages, and they can use builtin packages as dependencies (e.g. if your code uses MPI, you can just write a package that depends_on('mpi') and leverage the builtin MPI recipes.

Q: singularity remote servers. I have an air-gapped network. How can I make packages available from an 'internal' remote server. I cannot use the cloud versions mentioned.
I suppose I am asking for information about how to build an internal singularity 'remote' server so that I can add that to the server list using 'singularity remote add' command.
I cannot find information on how to do this or IF it is possible.
I know you mentioned the API, but this just tells me the syntax of commands. Is there a singularity 'service' that can be installed that supports setup of a remote server?
I'd ideally harvest the required packages from the public sites and then port them into the internal air-gapped environment. Hope this makes sense!!

A: think I found it. "sregistry" is the magic search term!
(https://singularityhub.github.io/sregistry-cli/getting-started#endpoints)

Q: Can you go over how we can access the slides again?