

Custom software

When the built-in software doesn't fit your needs, feel free to bring your software to the cluster. This article covers how you can do this in Lmod and containers and how to share it with your teammates.

Lmod

First, please study the official Lmod guide about [Personal Modulefiles](#). Then we recommend you place your software and modulefiles in the group scratch file set. Make sure to make all directories and files readable by your group. If you don't want your teammate to modify it, make it writable only by the owner.

Following is an example of compiling git 2.38.1 and adding it as a custom module:

```
# define where to put our software and modulefiles
MODHOME=/pfss/scratch02/appcara
PKGPATH=$MODHOME/pkg
MODPATH=$MODHOME/modulefiles

# download source code and compile
cd $MODHOME
wget https://mirrors.edge.kernel.org/pub/software/scm/git/git-2.38.1.tar.gz
tar xf git-2.38.1.tar.gz
cd git-2.38.1
./configure --prefix=$PKGPATH/git/2.38.1
make && make install

# setup the module file
mkdir -p $MODPATH/git
cat > $MODPATH/git/2.38.1.lua <<EOF
local home      = "/pfss/scratch02/appcara"
local version   = myModuleVersion()
local pkgName   = myModuleName()
local pkg       = pathJoin(home, "pkg", pkgName, version, "bin")
prepend_path("PATH", pkg)
EOF
```

Now everyone who has access to your group scratch directory can use your new module with the following commands.

```
# use the custom module path
module use /pfss/scratch02/appcara/modulefiles

# check if our git is available
module avail git

# load the module and test
module load git/2.38.1
git --version    # you should see git version 2.38.1
```

Containers

The cluster is using [Singularity](#). Containers are .sif files on the file system. You may extend ours, download from the internet or build your containers from scratch. Below lists a few ways to prepare software containers.

We recommend placing your custom images in the containers directory in your home or group scratch folder. So you and your teammate can see them in the web portal.

Pull from the internet

There are tons of container images on the internet. You may want to start by searching from some repositories:

- [Docker Hub](#)
- [Singularity Library](#)
- [NVIDIA GPU Cloud](#)
- [Quay.io](#)
- [BioContainers](#)

Below are some examples of pulling containers from the above public repositories.

```
# put in the containers folder so web portal can see them
mkdir ~/containers
cd ~/containers

# Singularity Hub
```

```
singularity pull rstudio.3.4.4.sif shub://mjstealey/rstudio

# Singularity Cloud Library
singularity pull alpine.3.15.3.sif library://alpine:latest

# Docker Hub
singularity pull julia.1.8.2.sif docker://julia:alpine3.16

# NVIDIA GPU Cloud
singularity pull pytorch.22.09-py3.sif docker://nvcr.io/nvidia/pytorch:22.09-py3
```

Extend a built-in image

Sometimes we may want to prepare our image. The following example shows how to extend the built-in PyTorch image by installing some python packages.

First, let's create a gym.def file to instruct singularity on how to build our new image.

```
BootStrap: localimage
From: /pfss/containers/pytorch.22.09-py3.sif

%post
    pip install gym==0.24.1 gym[atari,accept-rom-license]==0.24.1
    pip install atari-py==0.2.9 pybullet==3.2.5
```

We are simply leveraging the pytorch.22.09 image and install the Gym library from OpenAI for reinforcement learning studies.

If you want to learn more about how to customize your image. Please study the [official documentation](#).

Next, we will run the below commands to build the image.

```
singularity build gym.sif gym.def

# verify if our image is working
singularity exec gym.sif pip list

# move it to the containers folder, then we can run it in the web portal
mkdir -p ~/containers
mv gym.sif ~/containers
```

Revision #28

Created 4 November 2022 07:45:31 by Loki Ng

Updated 5 May 2023 15:34:47 by Loki Ng