

Singularity

Singularity

Basics of singularity

How to create a customized singularity image

How to use a custom image in jupyterhub

Singularity

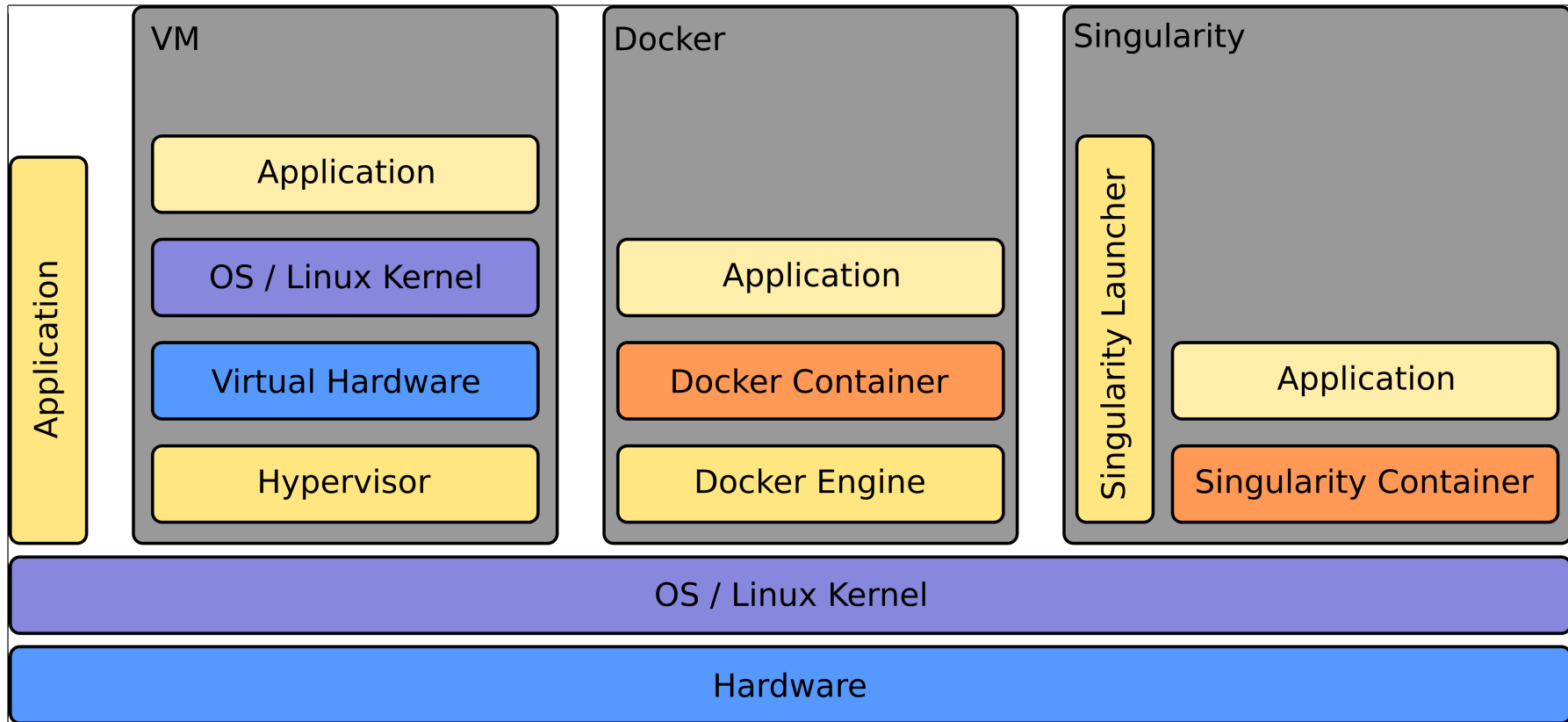
Singularity is a container solution.

Singularity supports HPC systems.

Singularity natively supports InfiniBand, Lustre, and resource managers (e.g. SLURM, SGE, etc.).

Singularity has built-in support for MPI and can access GPU resources.

Virtualization vs. Containerization



Singularity

basic commands

```
singularity pull  
singularity exec  
singularity shell  
singularity run  
singularity build
```

Singularity

basic examples

```
(skylake) [VSC-4]$ module load singularity/3.8.5-gcc-12.2.0-kdm7usg  
(zen3) [VSC-5]$ module load singularity/3.8.3-gcc-11.2.0-k6oedzn
```

```
(skylake/zen3) [VSC-4/5]$ singularity pull docker://ubuntu:latest  
(skylake/zen3) [VSC-4/5]$ singularity pull docker://centos:latest  
(skylake/zen3) [VSC-4/5]$ singularity pull docker://python:latest
```

Singularity

basic examples

```
(skylake/zen3) [VSC-4/5]$ singularity exec ubuntu_latest.sif cat /etc/lsb-  
release  
(skylake/zen3) [VSC-4/5]$ singularity exec centos_latest.sif cat /etc/redhat-  
release
```

Singularity

basic examples

```
(skylake/zen3) [VSC-4/5]$ singularity shell ubuntu_latest.sif
Singularity> cat /etc/lsb-release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=20.04
DISTRIB_CODENAME=focal
DISTRIB_DESCRIPTION="Ubuntu 20.04.3 LTS"
```


Singularity

basic examples

```
$ cat hello.py
import sys
print("Hello World: The Python version is %s.%s.%s" % sys.version_info[:3])
```

```
(skylake/zen3) [VSC-4/5]$ python3 hello.py
Hello World: The Python version is 3.9.5
```

```
(skylake/zen3) [VSC-4/5]$ singularity exec python_latest.sif python ./hello.py
Hello World: The Python version is 3.11.0
```

Singularity

basic examples

```
(skylake/zen3) [VSC-4/5]$ singularity shell python_latest.sif
Singularity> python ./hello.py
Hello World: The Python version is 3.11.0

Singularity> cat /etc/os-release
```

```
(skylake/zen3) [VSC-4/5]$ singularity run python_latest.sif
Python 3.10.0 (default, Oct 26 2021, 22:20:53) [GCC 10.2.1 20210110] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

basic job script examples on VSC

VSC-4:

```
#!/bin/bash
#
#SBATCH -J myjob
#SBATCH -o output.%j
#SBATCH -p skylake_0096
#SBATCH -N 1

module load singularity/3.8.5-gcc-12.2.0-kdm7usg
singularity exec docker://python:latest python ./hello.py
```

VSC-5:

```
#!/bin/bash
#
#SBATCH -J myjob
#SBATCH -o output.%j
#SBATCH -p zen3_0512
#SBATCH -N 1

module load singularity/3.8.3-gcc-11.2.0-k6oedzn
singularity exec docker://python:latest python ./hello.py
```

Singularity image for jupyterhub

Examples can be found in

```
/home/fs70824/traineeXX/VSC-Jupyterhub/singularity-examples
```

Minimal notebook with developer tools:

```
# cat jupyter-minimal.def
BootStrap: docker
From: jupyter/minimal-notebook:latest

%post
apt update
apt -y upgrade
apt -y install environment-modules gcc gfortran g++
/opt/conda/bin/pip install batchspawner
```

On a workstation with root permission:

```
singularity build jupyter-minimal.sif jupyter-minimal.def
```

Singularity image for jupyterhub

Minimal notebook with python modules:

```
# cat jupyter-pytorch.def
Bootstrap: docker
From: jupyter/minimal-notebook:latest

%post
/opt/conda/bin/pip install torch numpy matplotlib batchspawner
```

On a workstation with root permission:

```
singularity build jupyter-pytorch.sif jupyter-pytorch.def
```

Singularity image for jupyterhub

VIENNA SCIENTIFIC CLUSTER | jupyterhub Home Token

Server Options

Select profile to use:
VSC-4 (Singularity Image) for Training

Select IDE to start with: JupyterLab Jupyter Notebook

Choose image type: Use predefined image Use custom image

Provide your own custom singularity image
/home/fs70824/trainee78/VSC-Jupyterhub/singularity-examples/jupyter-minimal.sif

Please note: The image has to be placed on a share reachable from the cluster nodes


Number of (hyperthreading) cores
2 cores 4 cores 8 cores 12 cores 24 cores 48 cores

Memory needed
8GB 16GB 32GB 64GB

Maximum running time
4 hours 12 hours 24 hours 3 days

Start

Singularity image for jupyterhub

jupyterhub Home Token

Server Options

Select profile to use:

Select IDE to start with:

JupyterLab Jupyter Notebook

Choose image type:

Use predefined image Use custom image

Provide your own custom singularity image

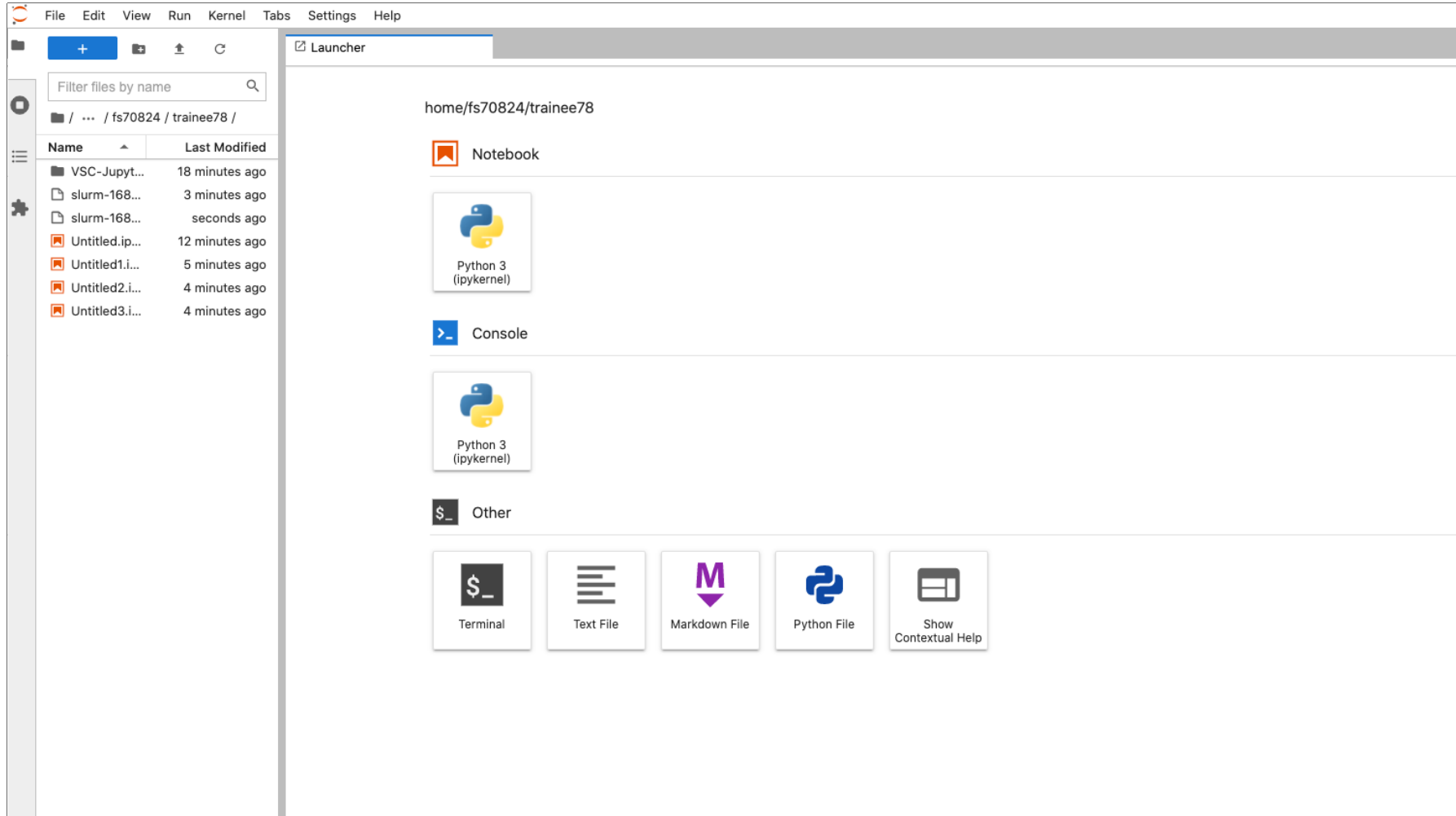
Please note: The image has to be placed on a share reachable from the cluster nodes

Number of (hyperthreading) cores

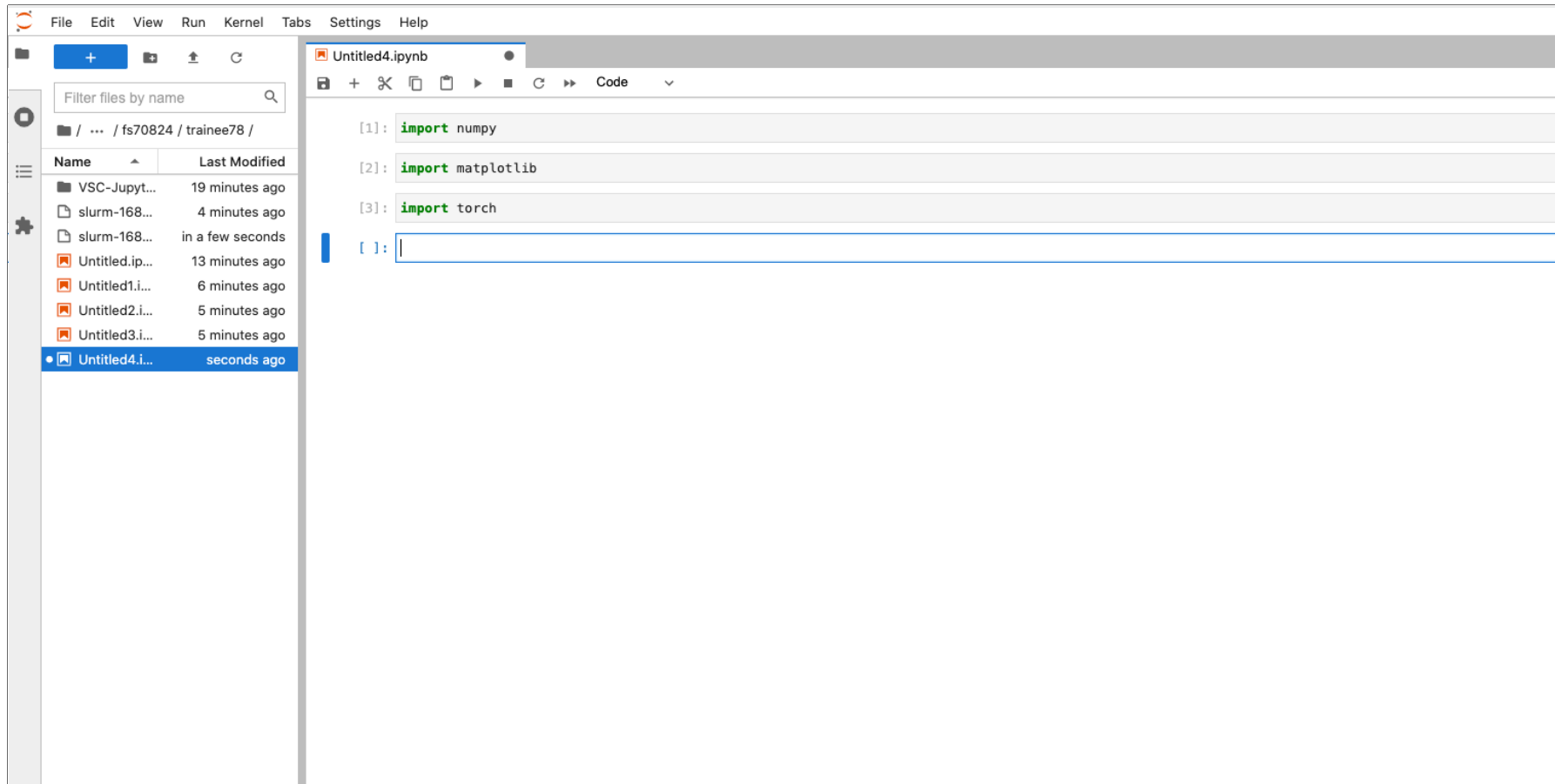
Memory needed

Maximum running time

Singularity image for jupyterhub



Singularity image for jupyterhub



Conda virtual env in jupyterhub

Advantage: no need to rebuild the container everytime you want to install a new package.

Conda virtual env in jupyterhub

Required steps:

- Create a venv
- Make the venv available in jupyterlab

Conda virtual env in jupyterhub

Create a new venv:

```
(skylake) [VSC-4]$ module load miniconda3/4.12.0-intel-2021.5.0-kwofrye  
(skylake) [VSC-4]$ conda create -n newenv python=3.10  
Collecting package metadata: done  
Solving environment: done
```

```
## Package Plan ##
```

```
environment location: /home/fs60000/jz/.conda/envs/newenv
```

```
added / updated specs:  
- python=3.10
```

The following packages will be downloaded:

```
...
```

Conda virtual env in jupyterhub

```
(skylake) [VSC-4]$ conda init bash
(skylake) [VSC-4]$ bash
(base)(skylake) [VSC-4]$ conda activate newenv
(newenv)(skylake) [VSC-4]$ which python3
~/conda/envs/newenv/bin/python3
(newenv)(skylake) [VSC-4]$ python3 --version
Python 3.10.0
```

Conda virtual env in jupyterhub

Add Virtual Environment to Jupyter Notebook:

```
(newenv)(skylake) [VSC-4]$ pip install --user ipykernel
Collecting ipykernel
  Using cached ipykernel-6.5.0-py3-none-any.whl (125 kB)
Collecting debugpy<2.0,>=1.0.0
  Downloading debugpy-1.5.1-cp310-cp310-
manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_12_x86_64.manylinux2010_x86_
64.whl (1.9 MB)
  |_____| 1.9 MB 27.1 MB/s
```

...

```
(newenv)(skylake) [VSC-4]$ python -m ipykernel install --user --name=newenv
Installed kernelspec newenv in /home/fs60000/jz/.local/share/jupyter/kernels
/newenv
```

Conda virtual env in jupyterhub

```
(newenv)(skylake) [VSC-4]$ cat /home/fs60000/jz/.local/share/jupyter/kernels/newenv/kernel.json
{
  "argv": [
    "/home/fs60000/jz/.conda/envs/newenv/bin/python",
    "-m",
    "ipykernel_launcher",
    "-f",
    "{connection_file}"
  ],
  "display_name": "newenv",
  "language": "python",
  "metadata": {
    "debugger": true
  }
}
```

Conda virtual env in jupyterhub

```
(newenv)(skylake) [VSC-4]$ jupyter kernelspec list
Available kernels:
myenv      /home/fs60000/jz/.local/share/jupyter/kernels/myenv
newenv     /home/fs60000/jz/.local/share/jupyter/kernels/newenv
python3    /home/fs60000/jz/.local/share/jupyter/kernels/python3
```


Conda virtual env in jupyterhub

The screenshot displays the JupyterLab interface. On the left is a file browser showing a directory structure under `/fs60000/jz/`. The main area is the 'Launcher' tab, which provides options for creating new content. The current path is `home/fs60000/jz`.

The Launcher is organized into three sections:

- Notebook:** Offers three Python environment options: `Python 3 (ipykernel)`, `myenv`, and `newenv`.
- Console:** Offers the same three Python environment options.
- Other:** Offers five options: `Terminal`, `Text File`, `Markdown File`, `Python File`, and `Show Contextual Help`.

Name	Last Modified
slurm-168...	3 days ago
slurm-168...	3 days ago
slurm-168...	a day ago
slurm-168...	28 minutes ago
slurm-168...	in a few seconds
SPEC-201...	2 years ago
SPECFP_V...	8 months ago
SPECFP_V...	8 months ago
specfp201...	2 years ago
specfp201...	2 years ago
Untitled.ip...	2 months ago
untitled.md	6 days ago
untitled.py	6 days ago
untitled.txt	6 days ago
Untitled1.i...	2 months ago
Untitled10....	6 days ago
Untitled11.i...	6 days ago
Untitled12....	6 days ago
Untitled13....	6 days ago
Untitled14....	6 days ago
Untitled15....	6 days ago
Untitled16....	3 days ago
Untitled17.i...	3 days ago

Conda virtual env in jupyterhub

Try:

```
(newenv)(skylake) [VSC-4]$ pip install numpy
```