



Finanziato
dall'Unione europea
NextGenerationEU



Ministero
dell'Università
e della Ricerca



Italiadomani

PIANO NAZIONALE
DI RIPRESA E RESILIENZA



Centro Nazionale di Ricerca in HPC,
Big Data and Quantum Computing

INFN



Centro Nazionale di Ricerca in HPC,
Big Data and Quantum Computing

Current High Rate analysis platform

Adele D'Onofrio, Gianluca Sabella, Bernardino Spisso, Tommaso Tedeschi
on behalf of WP5

Quasi-interactive analysis meeting - 23 January 2023 - Online

What we have now

- Resources:
 - k8s cluster:
 - master node with 8 CPUs, 16 MB RAM, 30 GB disk
 - 9 worker nodes with 16 CPUs, 33 MB RAM, 155 GB disk
- Entrypoint: <https://hub.192.135.24.49.myip.cloud.infn.it>
 - JupyterHub deployed with [WP5 JHub Helm chart](#)
 - Login via IAM DEMO: <https://iam-demo.cloud.cnaf.infn.it/>
 - Available ready-to-use JupyterLab images:
 - ghcr.io/ttedeschi/jlab:wp5-alma8-0.0.34 (default one, almalinux8 + python3.11 + Dask)
 - ghcr.io/ttedeschi/jlab:wp5-alma8-0.0.40 (almalinux8 + python3.11 + Dask + ROOT 6.30)
 - In both cases, a Dask cluster can be spawned on Kubernetes nodes via the Dask Labextension (for `ghcr.io/ttedeschi/jlab:wp5-alma8-0.0.40` Dask workers are deployed with the same ROOT version)

Access steps

Sign in with OAuth 2.0

Sign in with your iam-demo credentials

ttedesch

Password

Sign in

Forgot your password?

Or sign in with

Google

Not a member?

Apply for an account

Server Options

Select your desired image:

Select your desired number of cores: 1

Select your desired memory size: 2GB

Start

Launcher

Notebook

Python 3 (ipykernel) ROOT C++

Console

Python 3 (ipykernel) ROOT C++

Other

DASK DASHBOARD URL

Launcher

persistent-storage/analysis-examples/PyHEP2022

Notebook

Python 3 (ipykernel) ROOT C++

Console

Python 3 (ipykernel) ROOT C++

Other

CLUSTERS + NEW

Launcher Workers

CPU Use (%)

Memory Use (%)

name	address	nthreads	cpu	memory	limit	memory	managed	unmanag	unmanag	spilled	# fds	net read	net write	disk read	disk write
Total (10)		10	5%	1.4 GiB	20.0 GiB	7.1%	0.0	886.3 MiB	559.7 MiB	0.0	220	3 KiB	15 KiB	0	0
dask-itec tcp://10.4.1		4%	144.6 MiB	2.0 GiB	7.1%	0.0	64.8 MiB	79.8 MiB	0.0	22	286 B	1 KiB	0	0	
dask-itec tcp://10.4.1		4%	144.4 MiB	2.0 GiB	7.0%	0.0	64.9 MiB	79.5 MiB	0.0	22	286 B	1 KiB	0	0	
dask-itec tcp://10.4.1		6%	144.2 MiB	2.0 GiB	7.0%	0.0	144.2 MiB	64.0 KiB	0.0	22	286 B	1 KiB	0	0	
dask-itec tcp://10.4.1		6%	144.4 MiB	2.0 GiB	7.1%	0.0	144.4 MiB	56.0 KiB	0.0	22	286 B	1 KiB	0	0	
dask-itec tcp://10.4.1		6%	144.3 MiB	2.0 GiB	7.0%	0.0	64.8 MiB	79.5 MiB	0.0	22	285 B	1 KiB	0	0	
dask-itec tcp://10.4.1		6%	144.5 MiB	2.0 GiB	7.1%	0.0	144.3 MiB	148.0 KiB	0.0	22	549 B	2 KiB	0	0	
dask-itec tcp://10.4.1		6%	145.5 MiB	2.0 GiB	7.1%	0.0	64.5 MiB	80.9 MiB	0.0	22	286 B	1 KiB	0	0	
dask-itec tcp://10.4.1		4%	145.5 MiB	2.0 GiB	7.1%	0.0	65.0 MiB	80.5 MiB	0.0	22	286 B	1 KiB	0	0	
dask-itec tcp://10.4.1		4%	144.0 MiB	2.0 GiB	7.0%	0.0	64.8 MiB	79.3 MiB	0.0	22	287 B	1 KiB	0	0	
dask-itec tcp://10.4.1		6%	144.6 MiB	2.0 GiB	7.1%	0.0	64.6 MiB	80.0 MiB	0.0	22	285 B	1 KiB	0	0	

CLUSTERS + NEW

KubeCluster 2
Scheduler Address: tcp://dask-ttedesch-...
Dashboard URL: http://dask-ttedesch-...
Number of Cores: 10
Memory: 20.0 GiB
Number of Workers: 10

Launcher Workers

CPU Use (%)

Memory Use (%)

name	address	nthreads	cpu	memory	limit	memory	managed	unmanag	unmanag	spilled	# fds	net read	net write	disk read	disk write
Total (0)															

CLUSTERS + NEW

KubeCluster 2
Scheduler Address: tcp://dask-ttedesch-...
Dashboard URL: http://dask-ttedesch-...
Number of Cores: 0
Memory: 0 B
Number of Workers: 0

SCALE SHUTDOWN

```
from dask.distributed import Client
client = Client("tcp://dask-ttedesch-e8f2e00d-8-scheduler.jhub:8786")
client
```

CLUSTERS + NEW

KubeCluster 2
Scheduler Address: tcp://dask-ttedesch-...
Dashboard URL: http://dask-ttedesch-...
Number of Cores: 10
Memory: 20.0 GiB
Number of Workers: 10

Dask cluster deploy steps



How to apply for an account

If you haven't done already, apply for an account here

<https://iam-demo.cloud.cnaf.infn.it/>

with motivation: "spoke2 tests"

Register at iam-demo

This is the iam-demo registration page.

To proceed with the registration please fill in your personal information below.

Given name
Your first name

Family name
Your family name

Email
Your email address

Username
Choose a username

Notes
Providing a clear explanation on the motivation behind this request will likely speed up the approval process.

Register Reset Form

INDIGO IAM for iam-demo

iam-demo.cloud.cnaf.infn.it/dashboard#/home

IAM for iam-demo Tommaso Tedeschi

Tommaso Tedeschi iam-demo

Account Management

- Home
- My clients
- Client management
- MitreID Dashboard

Tommaso Tedeschi
ttedesch
16755a36-7b7a-4314-a5a4-009e92b59f8d

Email tommaso.tedeschi@pg1.infn.it

Status ✓

Created a month ago

Updated a month ago

End time N/A

Edit Details

Change Password

Groups

Group requests

No request found

Join a group

Linked accounts

No linked accounts found

Link external account

X.509 certificates

No certificates found

SSH keys

No keys found for user

Add ssh key

Attributes

No attributes found

once approved, request to be added to "highrate" group

search "highrate" (same motivations)

Join group(s)?

Select one or more groups

Only groups you're not already a member of or for which there's no pending request will be shown...

Type something...

Provide a motivation for your request(s)

This motivation will be show to the administrators that will manage your request

Explain why you want to be a member of group ...

Join group(s) Cancel

Once added to the highrate group (it should appear among your "Groups"), you are ready to go!

DEMO



https://drive.google.com/file/d/1xe7JnEDpsPZBICtl_krstdwg1aNZ4Xkf/view?usp=sharing

<https://github.com/ttedeschi/WP5-examples/tree/main>

Some important points

- At the moment **each user is assigned 10 GB of persistent storage**:
 - Anything that is stored outside the `persistent-storage` will be lost when the JLab session ends
 - No redundancy nor any production backup is in place for persistent storage:
BACKUP EVERYTHING ELSEWHERE! since data will be lost in case of a node failure
- **You can create your own image** of course. The suggested procedure is to start from the [Dockerfile](#) of the working images:
 - Make sure that Dask scheduler and workers are spawned with your custom image by modifying [this line](#) accordingly
- A monitoring dashboard setup is in progress
- Let us know if you need cvmfs
- Plan to document everything
- Everything is in beta at the moment, please report any issue or bug!