#### **Corso Formazione**

29 Set 2023

Igor Abritta



## 1. Intro

## Introduzione

Come già detto nella precedente presentazione, sulla INFN-Cloud dashboard si hanno a disposizione diversi 'tools' che possono essere utili per gli esperimenti.

In questa presentazione andremo a parlare di uno strumento in particolare:



INFN Cloud Dashboard Deployments - Advanced - External Links -CENTRALISED SERVICES: INFN Cloud object storage Notebooks as a Service **INFN Cloud Registry** (NaaS) 6 jupyter **ON-DEMAND SERVICES:** Virtual machine Docker-compose Run docker docke Spark + Jupyter cluster

Per creare una Coda HTCondor basta cliccare sul pulsante 'Configure' e procedere con le scelte opportune per il caso specifico.

Tra le varie opzioni possiamo trovare: Number of Slaves, Flavor del Slave e del Master e Immagine della VM





HTCondor cluster	
Description: Deploy a complete HTCondor cluster	
Deployment description	
description	
General IAM integration Advanced	
admin_token	
Enter your password	]
token for accessing K8s dashboard and Grafana admin password	1
number_of_slaves	
3	
number of K8s node VMs	
wn_image	
dodasts/cygno-wn	
HTCondor WN image name	
wn_tag	
	]
HTCondor WN tag name	
master_flavor	
Select	]
Number of VCPUs and memory size of the k8s master VM	
node_flavor	
Select	]
Number of VCPUs and memory size of each k8s node VM	
Submit O Cancel	

## 2. HTCondor

#### Cosa è HTCondor

HTCondor è un sistema di gestione di risorse distribuite e una piattaforma di gestione del lavoro utilizzata principalmente in ambienti di calcolo ad alta prestazione (HPC) e di calcolo distribuito.

HTCondor Manual https://htcondor.readthedocs.io/en/latest/



Dopo configurato:

**\$condor\_status** è uno comando versatile che può essere utilizzato per monitorare e interrogare il pool HTCondor.

[root@0aa14b701892 abrittac]#	condor_sta	tus					
Name	OpSys	Arch	State	Activity	LoadAv	Mem	ActvtyTime
slot1@wn-pod-5c986df8b4-5snh4	LINUX	X86_64	Unclaimed	Idle	0.000	2000	57+13:49:34
slot1@wn-pod-5c986df8b4-64fqd	LINUX	X86_64	Unclaimed	Idle	0.000	2000	57+09:17:39
slot1@wn-pod-5c986df8b4-c85km	LINUX	X86_64	Unclaimed	Idle	0.000	2000	57+13:48:11
slot1@wn-pod-5c986df8b4-gjnrs	LINUX	X86_64	Unclaimed	Idle	0.000	2000	57+13:48:20
slot1@wn-pod-5c986df8b4-h87dx	LINUX	X86_64	Unclaimed	Idle	0.000	2000	57+13:48:11
slot1@wn-pod-5c986df8b4-mwtjm	LINUX	X86_64	Unclaimed	Idle	0.000	2000	57+13:47:49
slot1@wn-pod-5c986df8b4-p9v29	LINUX	X86_64	Unclaimed	Idle	0.000	2000	57+13:48:24
slot1@wn-pod-5c986df8b4-q28lh	LINUX	X86_64	Unclaimed	Idle	0.000	2000	57+13:48:39
slot1@wn-pod-5c986df8b4-s67mr	LINUX	X86_64	Unclaimed	Idle	0.000	2000	57+13:48:16
slot1@wn-pod-5c986df8b4-trfnt	LINUX	X86_64	Unclaimed	Idle	0.000	2000	57+13:48:39

	Total	Owner	Claimed	Unclaimed	Matched	Preempting	Backfill	Drain	
X86_64/LINUX	10	0	0	10	0	0	0	0	
Total root@0aa14b70	10 1892 al	0 pritta	0 ]#[]	10	0	0	0	0	

**\$condor\_q -all** può essere utilizzato per guardare le info dei Job.

[root@0aa14b701892 abrittac]# condor\_q -all

Sche	edd: 131.154	.96.173.myip.d	cloud.ir	fn.it :	<131.	154.96	.173:31618?	@ 09/29/23	10:14:34
OWNER	BATCH_NAME	SUBMITTED	DONE	RUN	IDLE	TOTAL	JOB_IDS		
condor	ID: 7209	8/2 21:00		_	1	1	7209.0		
condor	ID: 8673	9/29 09:55	_	_	_	1	8673.0		

Total for query: 2 jobs; 1 completed, 0 removed, 1 idle, 0 running, 0 held, 0 suspended Total for all users: 2 jobs; 1 completed, 0 removed, 1 idle, 0 running, 0 held, 0 suspended

[root@0aa14b701892 abrittac]# []

#### \$condor\_q --better-analyse <JobID>

può essere utilizzato per avere più info su stato del job specifico. [root@0aa14b701892 abrittac]# condor\_q --better-analyse 8673

-- Schedd: 131.154.96.173.myip.cloud.infn.it : <131.154.96.173:31618?... The Requirements expression for job 8673.000 is

(TARGET.Arch == "X86\_64") && (TARGET.OpSys == "LINUX") && (TARGET.Disk >= RequestDisk) && (TARGET.Memory >= RequestMemory) && (TARGET.HasFileTransfer)

Job 8673.000 defines the following attributes:

DiskUsage = 12 ImageSize = 1 MemoryUsage = ((ResidentSetSize + 1023) / 1024) RequestDisk = DiskUsage RequestMemory = ifthenelse(MemoryUsage =!= undefined,MemoryUsage,(ImageSize + 1023) / 1024) ResidentSetSize = 0

The Requirements expression for job 8673.000 reduces to these conditions:

SlotsStepMatchedCondition[0]10TARGET.Arch == "X86\_64"[1]10TARGET.OpSys == "LINUX"[3]10TARGET.Disk >= RequestDisk[5]10TARGET.Memory >= RequestMemory[7]10TARGET.HasFileTransfer

8673.000: Job is completed.

\$condor\_submit -spool <file>

Per sottomettere un job sulla coda configurata.

[root@0aa14b701892 HTCondor]# condor\_submit -spool submit Submitting job(s). 1 job(s) submitted to cluster 8674. [root@0aa14b701892 HTCondor]#

\$condor\_transfer\_data <JobID>

Per riprendere l'output del job ed anche i file di .out, .log e .error

[root@0aa14b701892 HTCondor]# condor\_transfer\_data 8674
Fetching data files...

## 3. Hands-on

- 1 Entrare dentro Jupyter e cercare la cartella /jupyter-workspace/shared/abrittac/HTCondor/
- 2 Copiare il folder dentro la vostra cartella private

3 - Configurare/Creare il vostro .bashrc (nel folder **/cloud-storage/<username>**) utilizzando come esempio il bashrc.txt disponibile dentro la cartella che avete appena copiato.

```
1 # config condor CYGNO queue
2 cat > /etc/condor/condor_config.local << EOF
3 AUTH_SSL_CLIENT_CAFILE = /etc/pki/ca-trust/source/anchors/htcondor_ca.crt
4 SCITOKENS_FILE = /tmp/token
5 SEC_DEFAULT_AUTHENTICATION_METHODS = SCITOKENS
6 #Coda 10
7 COLLECTOR_HOST = 131.154.96.173.myip.cloud.infn.it:30618
8 SCHEDD_HOST = 131.154.96.173.myip.cloud.infn.it
9 EOF
```

4 - Fare un source del bashrc oppure aprire un altro terminale

5 - Girare il comando **\$condor\_status** per controllare che la configurazione della code è eseguita bene.

#### 6 - Dentro il folder abbiamo 3 altre file:

- **test\_condor.py:** Algoritmo molto semplice che crea un array e lo salva in .npy
- **exec\_reco.sh:** File che contiene la righa per fare girare l'algoritmo python
- **submit:** Esempio di file di sottomissione di condor

condor\_submit -spool submit

#### 6 - Dentro il folder abbiamo 3 altre file:

- **submit:** Esempio di file di sottomissione di condor

1	universe = vanilla
2	executable = /jupyter-workspace/shared/abrittac/HTCondor/exec_reco.sh
3	log = test_condor.log
4	<pre>output = test_condor.out</pre>
5	error = test_condor.error
6	
7	<pre>should_transfer_files = YES</pre>
8	when_to_transfer_output = ON_EXIT
9	
10	<pre>transfer_input_files = /jupyter-workspace/shared/abrittac/HTCondor/test_condor.py</pre>
11	<pre>transfer_output_files = dati_random.npy</pre>
12	
13	+OWNER = "condor"
14	queue

#### 6 - Dentro il folder abbiamo 3 altre file:

- **submit:** Esempio di file di sottomissione di condor

```
1 universe = vanilla
2 executable = /jupyter-workspace/shared/abrittac/HTCondor/exec_reco.sh
3 log = test_condor.log
4 output = test_condor.out
5 error = test_condor.error
6
7 should_transfer_files = YES
8 when_to_transfer_output = ON_EXIT
9
10 transfer_input_files = /jupyter-workspace/shared/abrittac/HTCondor/test_condor.py
11 transfer_output_files = dati_random.npy
12
13 +OWNER = "condor"
14 queue
```

#### 6 - Dentro il folder abbiamo 3 altre file:

- **submit:** Esempio di file di sottomissione di condor

```
1 universe = vanilla
2 executable = /jupyter-workspace/shared/abrittac/HTCondor/exec_reco.sh
3 log = test_condor.log
4 output = test_condor.out
5 error = test_condor.error
6
7 should_transfer_files = YES
8 when to transfer output = ON EXIT
9
10 transfer_input_files = /jupyter-workspace/shared/abrittac/HTCondor/test_condor.py
11 transfer_output_files = dati_random.npy
12
13 +OWNER = "condor"
14 queue
```

- 7 Sistemare il file **submit** mettendo li i folder giusto :
- 8 Finalmente: Sottomettere il job:

#### condor\_submit -spool submit

- 9 Farvi attenzione a il Job\_ID per identificare il vostro Job.
- 10 Monitorare il Job con il comando **condor\_q -all** o **condor\_q --better-analyse <JobID>**
- 11 Quando finito riprendere l'output con il comando **condor\_transfer\_data <JobID>**
- 12 Controllare se adesso avete i 4 file di output nella cartella (.npy, .out, .error, .log)

## 4. Tips & Tricks

### **Tips & Tricks**

- 1 Attenzione al size del folder inviato a Condor
  - Principalmente se è un folder versionato da Git

2 - Attenzione alla quantità di job in queue, la spool di condor nella nostra cloud by default ha un storage di 20Gb.

- I file per girare il job vengono cancellati solo dopo il **condor\_data\_transfer**
- 3 Refresh Token

Grazie