



ESCAPE

European Science Cluster of Astronomy &
Particle physics ESFRI research Infrastructures

CMS report DAC21

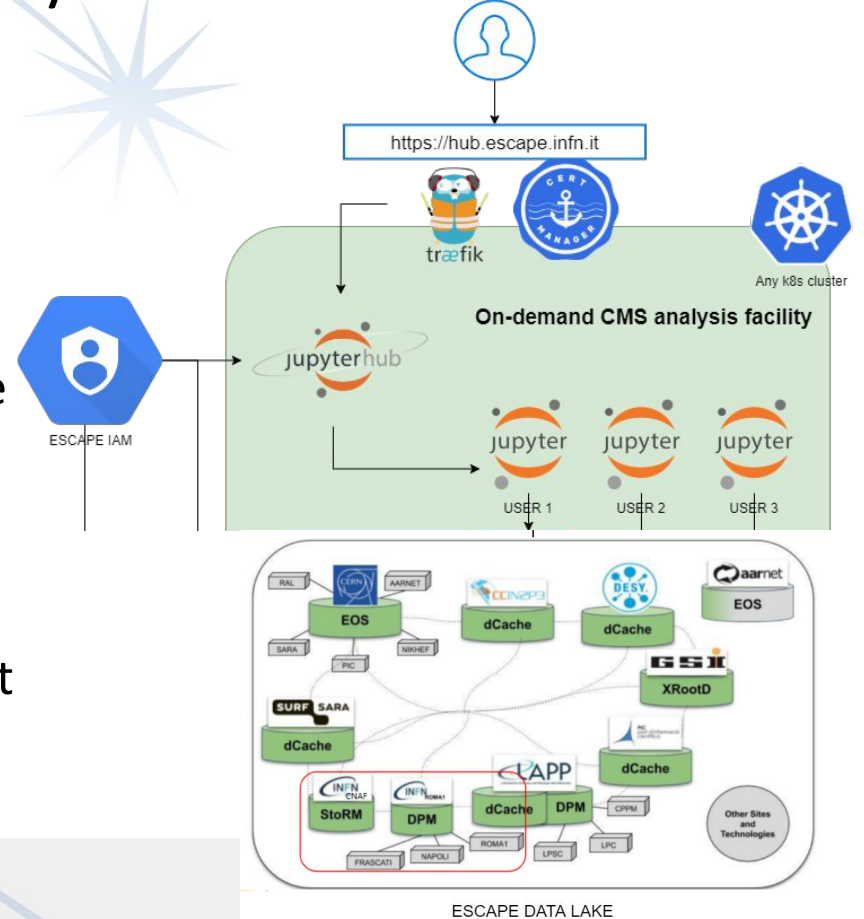
D. Ciangottini on behalf of ESCAPE-CMS team



RECAP: enabling CMS analysis on NanoAOD

Brief recap on the main objectives for CMS:

- **NanoAOD opendata O(100GB) → plain ROOT files** → simple python **ROOTDataFrame** as framework
- We are interested in replicating a simple analysis on a JupyterLab instance on a dedicated JupyterHUB hosted at CNAF
 - Also scaling out to a batch system
- We target opendata, but, of course, in terms of volume embargoed will be a bit more... “realistic”



Input registration

- Uploaded files in CMS_INFN_DCIANGOT scope via rucio upload:
 - Both X509 and **token** auth test with success
- Collected all the file in a dataset with a replica rule to CNAF

No problem to report, and none expected after the preparatory exercises

Name	Account	RSE Expression	Creation Date	Remaining Lifetime	State
CMS_INFN_DCIANGOT:ESCAPE-CMS-Opdata	dclangot	CNAF-STORM	2021-12-03T08:58:26.000Z	90d	OK

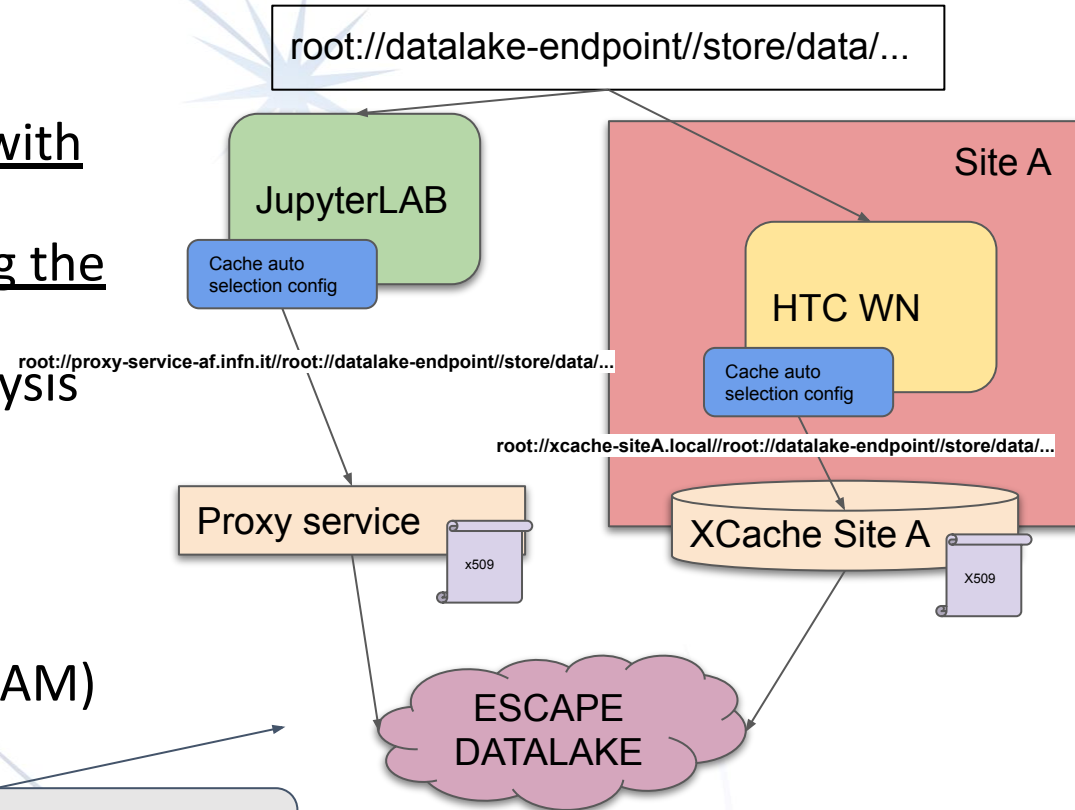
Name	Account	RSE Expression	Creation Date	Remaining Lifetime	State
CMS_INFN_DCIANGOT:Run2012C_DoubleElectron.root	dclangot	DESY-DCACHE	2021-11-15T09:27:26.000Z	4d	OK
CMS_INFN_DCIANGOT:ZZTo4mu.root	dclangot	DESY-DCACHE	2021-11-15T09:22:13.000Z	4d	OK
CMS_INFN_DCIANGOT:ZZTo4e.root	dclangot	DESY-DCACHE	2021-11-15T09:21:52.000Z	4d	OK
CMS_INFN_DCIANGOT:ZZTo2e2mu.root	dclangot	DESY-DCACHE	2021-11-15T09:21:26.000Z	4d	OK
CMS_INFN_DCIANGOT:SMHiggsToZZTo4L.root	dclangot	DESY-DCACHE	2021-11-15T09:20:34.000Z	4d	OK
CMS_INFN_DCIANGOT:Run2012C_DoubleMuParked.root	dclangot	DESY-DCACHE	2021-11-15T09:16:30.000Z	4d	OK
CMS_INFN_DCIANGOT:Run2012B_DoubleElectron.root	dclangot	DESY-DCACHE	2021-11-15T09:11:36.000Z	4d	OK
CMS_INFN_DCIANGOT:Run2012B_DoubleMuParked.root	dclangot	DESY-DCACHE	2021-11-15T09:09:17.000Z	4d	OK



Access data via a group-managed Cache

- The cache server authenticates with a “service” proxy with the lake
- It serves the client either trusting the ip or with experiment IAM
- In other words, we perform analysis tests for which the user could in principle forget about any configuration to access the data (once is authenticated via a JupyterHUB and an experiment IAM)

SETUP DEPLOYED AT CNAF
and used for the following tests



Analysis workflow on a notebook

- A simple CMS analysis has been performed via a notebook
- **RUCIO data discovery worked smoothly with authN via token** (seamless for the user)

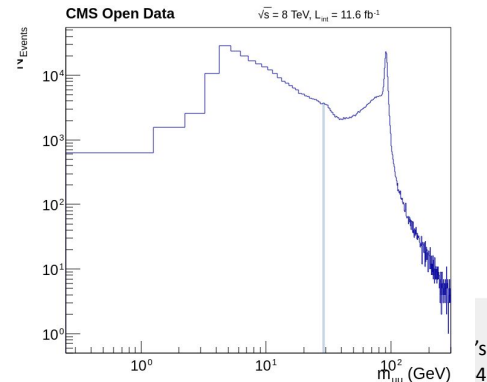
```
[7]: file_list = []
for repl in cli.list_replicas({"scope":"CMS_INFN_DCANGOT", "name": "ESCAPE-CMS-Opedata"}, schemes=["davs"]):
    file_list.append(list(repl["pfns"].keys())[0])

print(file_list)
```

```
['davs://dcache-door-doma01.desy.de:2880//escape/wp2_rucio_testbed/desy_dcache/CMS_INFN_DCANGOT/72/27/Run2012B_DoubleElectron.root', 'davs://dcache-door-doma01.desy.de:2880//escape/wp2_rucio_testbed/desy_dcache/CMS_INFN_DCANGOT/df/47/Run2012B_DoubleMuParked.root', 'davs://dcache-door-doma01.desy.de:2880//escape/wp2_rucio_testbed/desy_dcache/CMS_INFN_DCANGOT/d5/4a/Run2012C_DoubleElectron.root', 'davs://dcache-door-doma01.desy.de:2880//escape/wp2_rucio_testbed/desy_dcache/CMS_INFN_DCANGOT/1b/77/Run2012C_DoubleMuParked.root', 'davs://dcache-door-doma01.desy.de:2880//escape/wp2_rucio_testbed/desy_dcache/CMS_INFN_DCANGOT/64/e4/SMHiggsToZZTo4L.root', 'davs://dcache-door-doma01.desy.de:2880//escape/wp2_rucio_testbed/desy_dcache/CMS_INFN_DCANGOT/0b/5f/ZZTo2e2mu.root', 'davs://dcache-door-doma01.desy.de:2880//escape/wp2_rucio_testbed/desy_dcache/CMS_INFN_DCANGOT/42/5d/ZZTo4mu.root']
```

And so the read of those files:

- Direct webdav access via token
- Group cache read (ip trusted see previous slide)



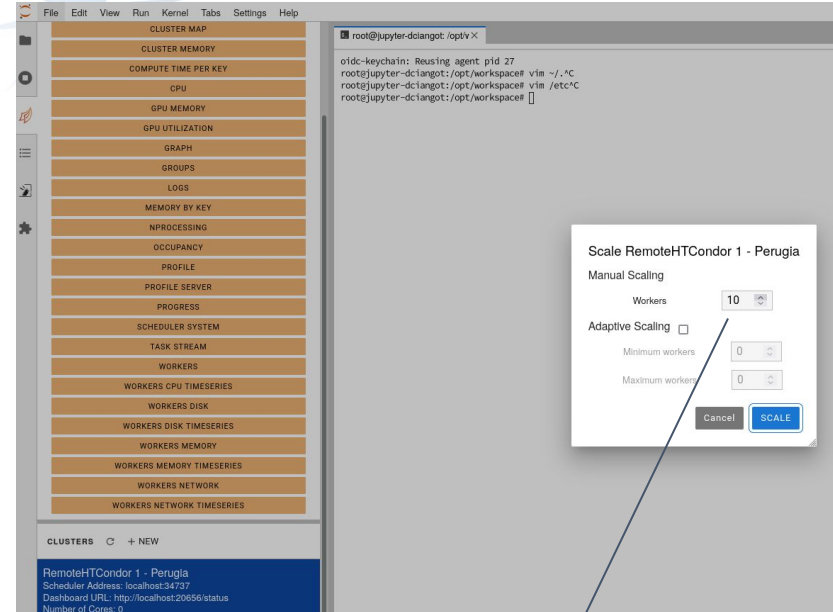
Analysis workflow on a batch system (HTCondor)

A similar exercise has been replicated using an HTCondor batch system deployed at CNAF (with some nodes federated in Perugia) where the **authN/Z is managed via ESCAPE-IAM as well.**

We tried 2 methods:

- the submission of an analysis job (legacy approach) - *no problem to report*
- Distribution of a notebook payload via DASK - *no problem to report*

The data access in this tests where only via cache server (no need to propagate credentials although it is possible to)



The screenshot shows the HTCondor web interface. On the left, there is a sidebar with various metrics like CLUSTER MAP, CLUSTER MEMORY, COMPUTE TIME PER KEY, CPU, GPU MEMORY, GPU UTILIZATION, GRAPH, GROUPS, LOGS, MEMORY BY KEY, NPROCESSING, OCCUPANCY, PROFILE, PROFILE SERVER, PROGRESS, SCHEDULER SYSTEM, TASK STREAM, WORKERS, WORKERS CPU TIMESERIES, WORKERS DISK, WORKERS DISK TIMESERIES, WORKERS MEMORY, WORKERS MEMORY TIMESERIES, WORKERS NETWORK, and WORKERS NETWORK TIMESERIES. On the right, there is a terminal window showing a shell prompt and some command output. A dialog box titled 'Scale RemoteHTCondor 1 - Perugia' is open, showing 'Manual Scaling' with 'Workers' set to 10. The dialog also has 'Adaptive Scaling' checked, 'Minimum workers' set to 0, and 'Maximum workers' set to 0. There are 'Cancel' and 'SCALE' buttons.

```
root@jupyter-dciangot:/opt/workspace# condor_q -totals
```

```
-- Schedd: 131.154.96.124.myip.cloud.infn.it : <131.154.96.124:31018?... @ 12/03/21 09:13:11
Total for query: 22 jobs; 11 completed, 0 removed, 1 idle, 10 running, 0 held, 0 suspended
Total for dciangot: 22 jobs; 11 completed, 0 removed, 1 idle, 10 running, 0 held, 0 suspended
Total for all users: 39 jobs; 28 completed, 0 removed, 1 idle, 10 running, 0 held, 0 suspended
```



Output registration

Finally, what did we do with the outputs:

- We were mainly interested in verify the rucio upload thing, that worked as expected
 - Not a surprise since we tested it also for the data injection
- In this regard we might have some room for improvements/investigation (see last slide)
 - In particular to automatize the staging to the lake without the user to specify “rucio upload”



Embargoed data

We also managed (credits to A.Ceccanti and L. Morganti) to give a first try to the upload of EMBARGOED CMS data to the CNAF-STORM RSE:

- Basically we succeeded to give **exclusive access** to the pfn reserved to the scope **CMS_EMBARGOED_DATA to the escape/cms users**
- The authorization is based on the user group both in the x509 and token!

In other words, all we needed to do was getting a valid x509 proxy or jwt with the voms/wlcg.groups attributes and then something like:

```
rucio upload --scope CMS_EMBARGOED_DATA --rse CNAF-STORM  
--lifetime 90000 --summary --name ZZTo4mu.root ZZTo4mu.root
```

Configuration for other interested sites can be replicated,
feel free to reach out!



Conclusions

Overall a **successful experience**, thanks to all the parties involved!

We have no outstanding issues to report.

We would like to look at:

- *How can RUCIO manage replication of “embargoed” data?*
- *Do we have any experience with ephemeral RSEs for buffer/staging areas?*
 - We might be interested in investigate further a similar solution for user analysis outputs



Final note on HPC

HPC tests are taking more than expected, I will try to get a summary of the situation for early next year at this point.

Still we demonstrated in the past that all we did in DAC21 can be reproducible with no major issues expected

