Istanziazione e utilizzo di batch system on demand su infrastrutture Cloud. L'esempio pilota dell'esperimento AMS Welcome and Introduction

Daniele Spiga spiga@infn.it

eosc-hub.eu

@EOSC_eu

Dissemination level: Public/Confidential *If confidential, please define:*
Disclosing Party: (those disclosing confidential information)
Recipient Party: (to whom this information is disclosed, default: project consortium)

EOSC-hub receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 777536.

Outline

Intro:

- some history and the preamble of EU project

Quick overview of AMS02 and its computing

Integration activities

- motivations and objectives
- what we did

Results and future

Some history, back in 2017…

The early stage of the presented activity was in the context of the project INDIGO-DataCloud.

- Just thoughts, discussions and ideas

We started real activity in the context of EOSC-hub project (2018)

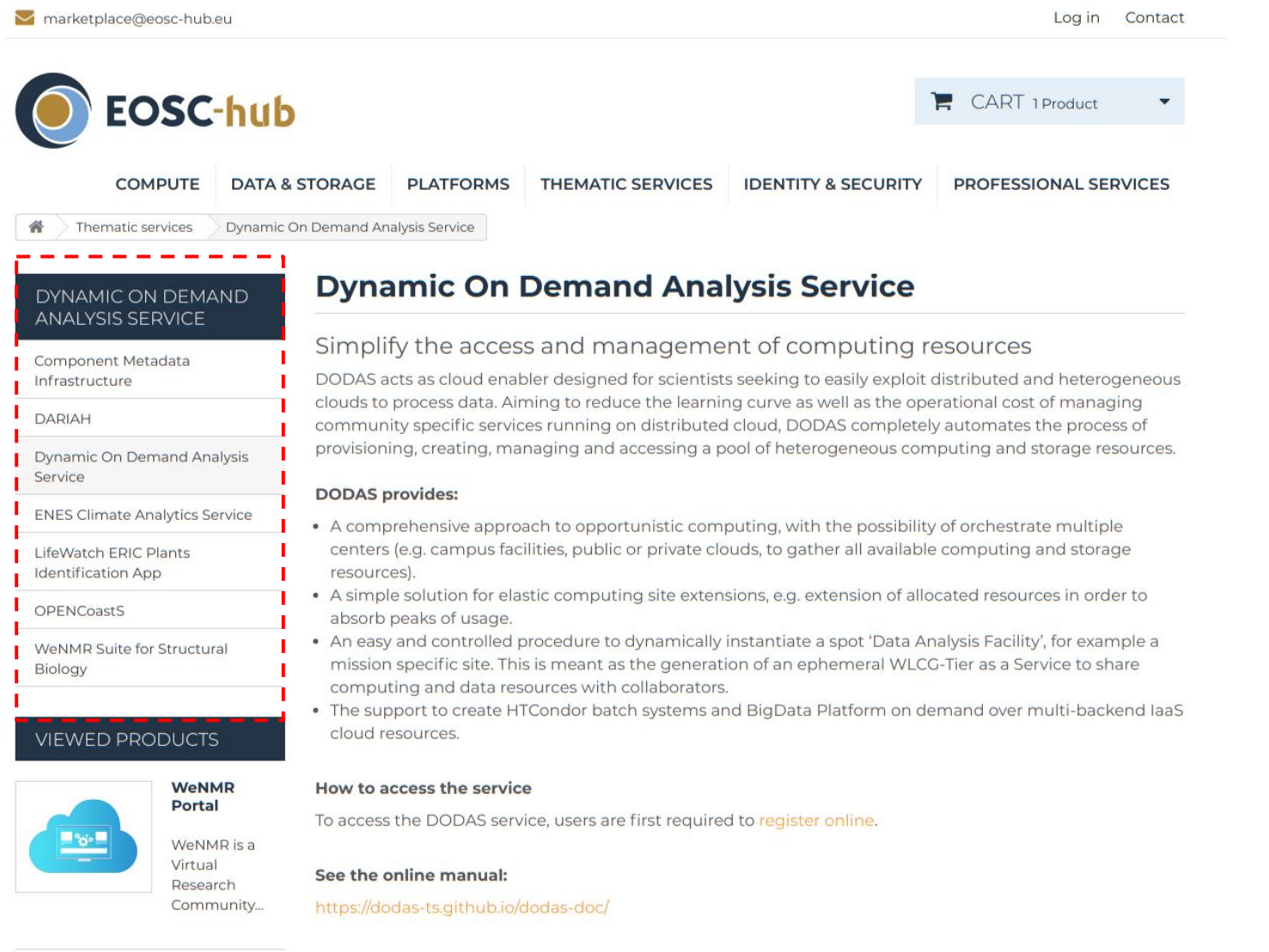
- in the framework of DODAS Thematic Service (see later)
  - one of the main objective was (and is) the exploitation towards new communities
  -

Thematic Services of EOSC-hub

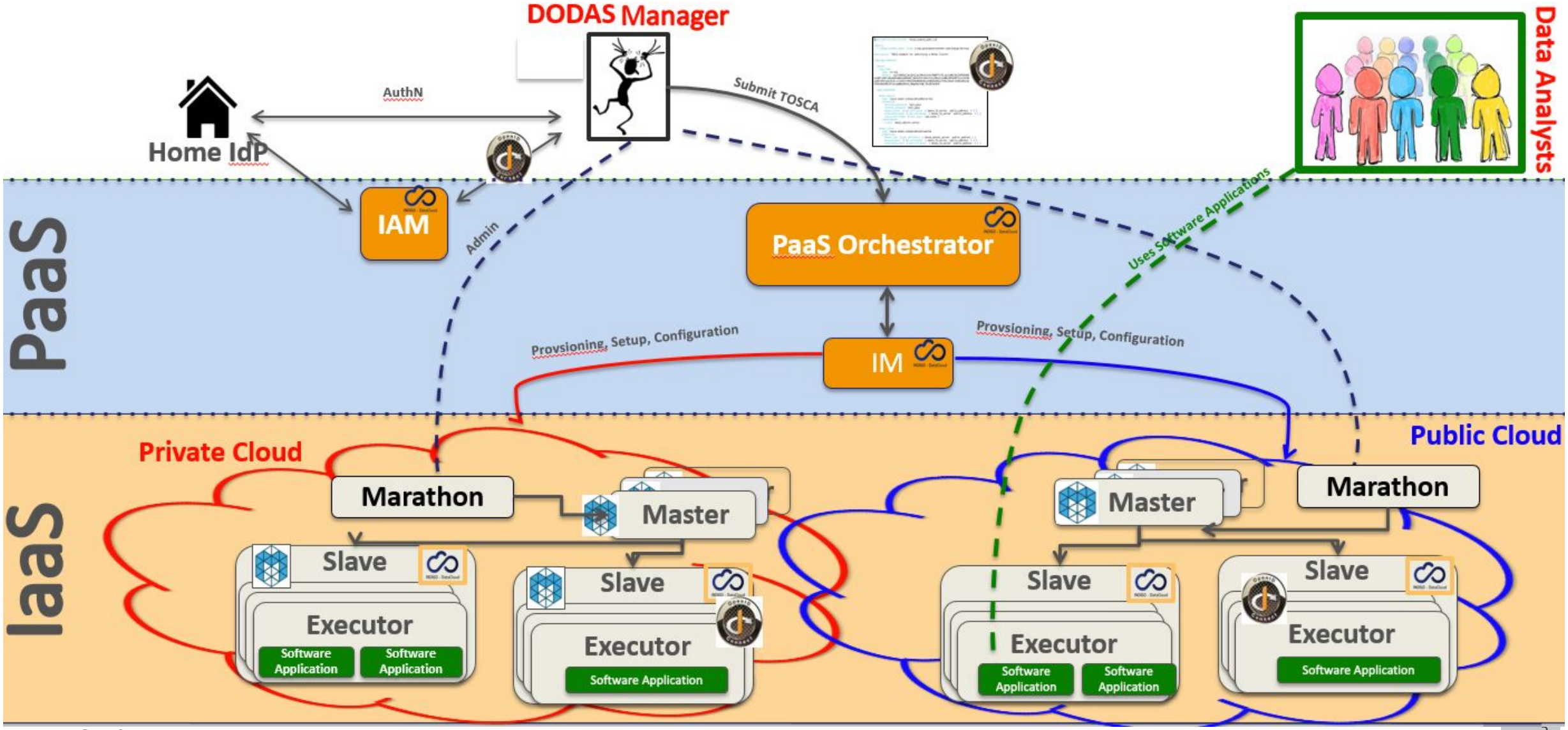
Thematic services of EOSC-hub are service which come from external partners willing to collaborate with the project.

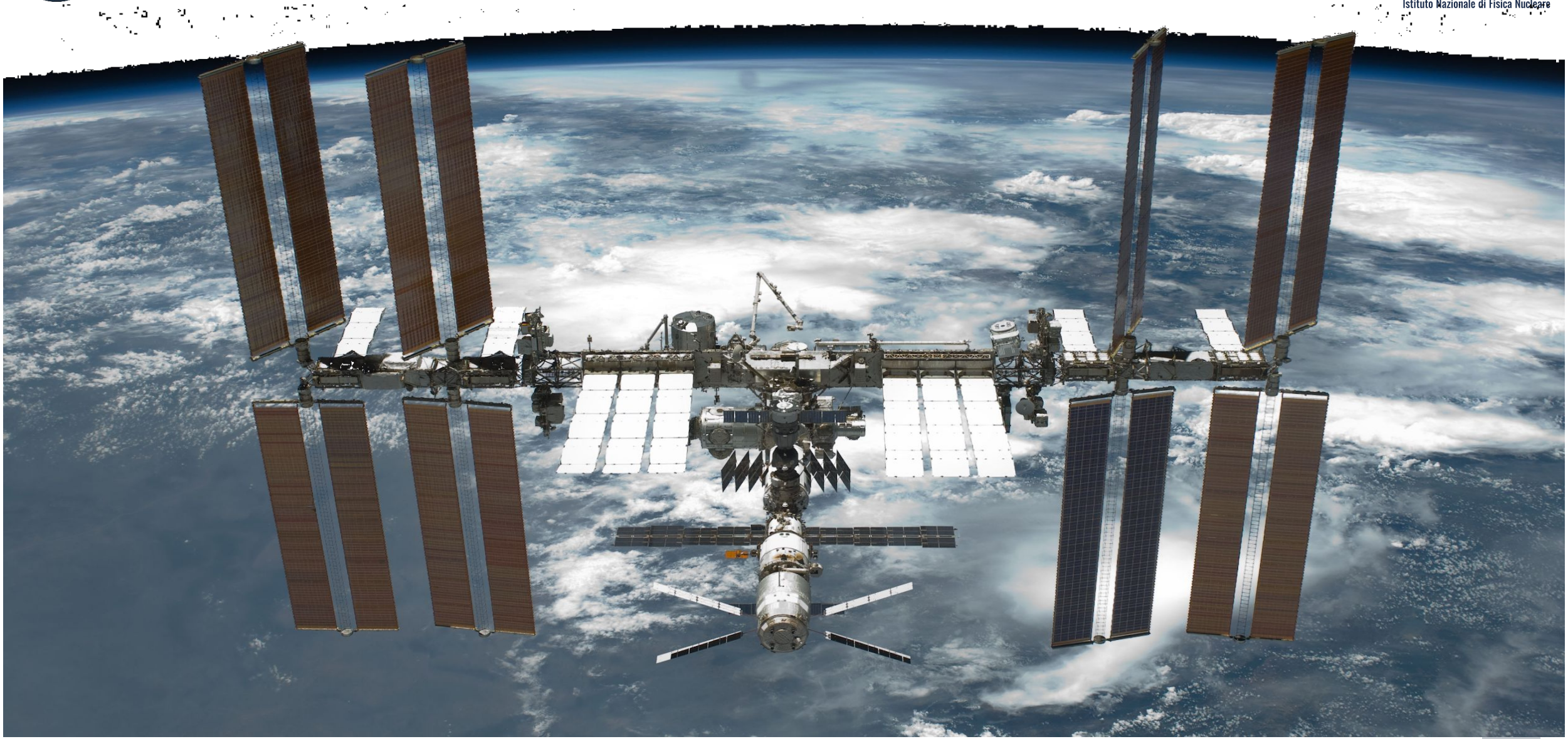
- The project has the objective of involving early adopters, stakeholder etc… aiming at reaching out to new user groups and service providers.

DODAS is one of the TS



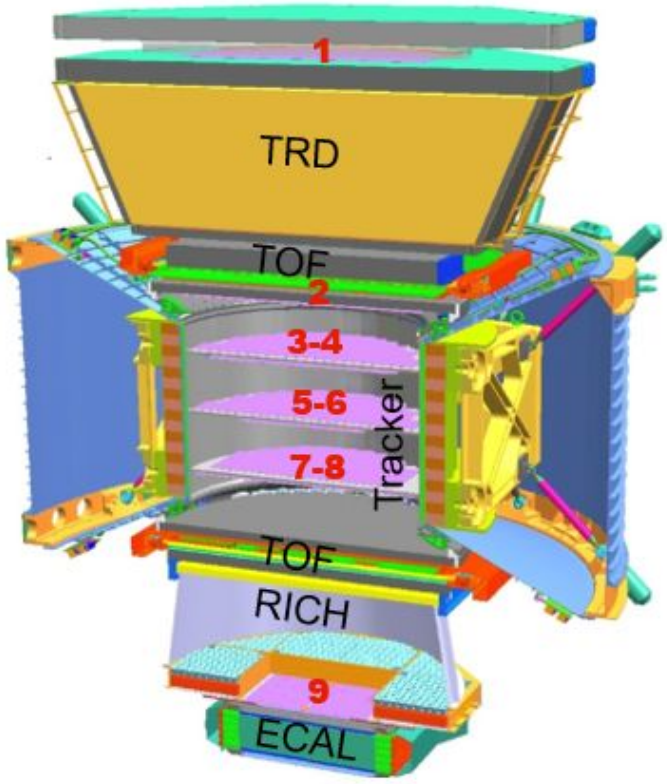
Reminder: What is DODAS in a nutshell





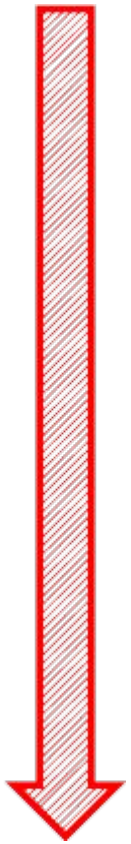
The AMS Experiment and AMS-02 detector

- installed on the International Space Station, ISS, on May 19, 2011
- operations 24h/day, 365d/year, since the installation
- 300k readout channels + 1500 temperature sensors
- acquisition rate up to 2kHz
- more than 600 microprocessors to reduce the rate from 7 Gb/s to 10 Mb/s
- 4 Science Runs (DAQ start/stop + calibration) per orbit: 1 Science Run = ~ 23 minutes of data taking
- on May 2019, ~135 billion triggers acquired
- 35 TB/year of raw data



courtesy of Matteo Duranti

Data Flow



- Data collected by AMS (~ 35 TB/year) are transmitted from the International Space Station, through the NASA infrastructure, to CERN
  - Copied at INFN-CNAF (master copy on tape)
- Reconstruction step, called std, produces ~ 100 TB/year
  - About a reconstruction step/year due to improved condition (which in turn is due to improved detector understanding)
    - example: last reprocessing required > 400 CPU-Years, to process > 5 years of AMS data taking
- Data reconstruction performed on regional centers distributed among: Europe, China and USA
  - **workflow is adapted, by hand, to the various regional centers**
  - job submission is managed by the LSF or PBS batch systems

Some more numbers about computing

- Analysis groups skim the reconstructed data producing ntuples
  - typically a production step takes about 10 CPU-Years
- AMS Scientists generate also huge amount of Monte Carlo datasets
  - as example in 2015 AMS used about 8000 CPU-Years, in 2016 about 11000

- Moreover the expectation is that these numbers will increase as far as the experiment will collect data
  - improving the understanding of the detector newer and updated Monte Carlo are required
    - statistics must follow the increased statistics of the real data

AMS resources landscape

- Stable, massive, resources:
  - CERN
  - CNAF
  - etc…
- Additional “stable” resources:
  - ASI
  - Small campus facilities (e.g "farm di Sezione")
- Temporary "free" resources: Opportunistic Computing
  - Any grant e.g. User grant and/or Funding Agency grant
  - Computing center @ China
  - Any provider from any partner of the collaboration

Thoughts…

Credits Matteo Duranti

- AMS computing model has not originally designed to cope with cloud computing paradigm
- AMS software is not automatically portable
- Costs for deploying AMS Environment on new resources might be to high
  - setting-up environment requires huge effort wrt obtained gain
  - keep in mind the limited effort of personnelle fully committed on computing
- it does not result trivial to exploit modern computing infrastructures etc

- All this appears to be a limiting factor in the scope of the data analyst daily work
  - the value is not in setting up computing…

A Gap to fill (the objective)

Need a technology to enable the following:

- Generation of on-demand batch systems for data processing
  - User friendly solution to exploit "any cloud platform"
- Exploitation of opportunistic computing
  - A solution to manage stateless sites
    - Resources w/o permanently dedicated support for a specific experiment and/or activity;
- Elastic extension of existing facilities/batches
  - e.g To absorb peaks of resource usage;

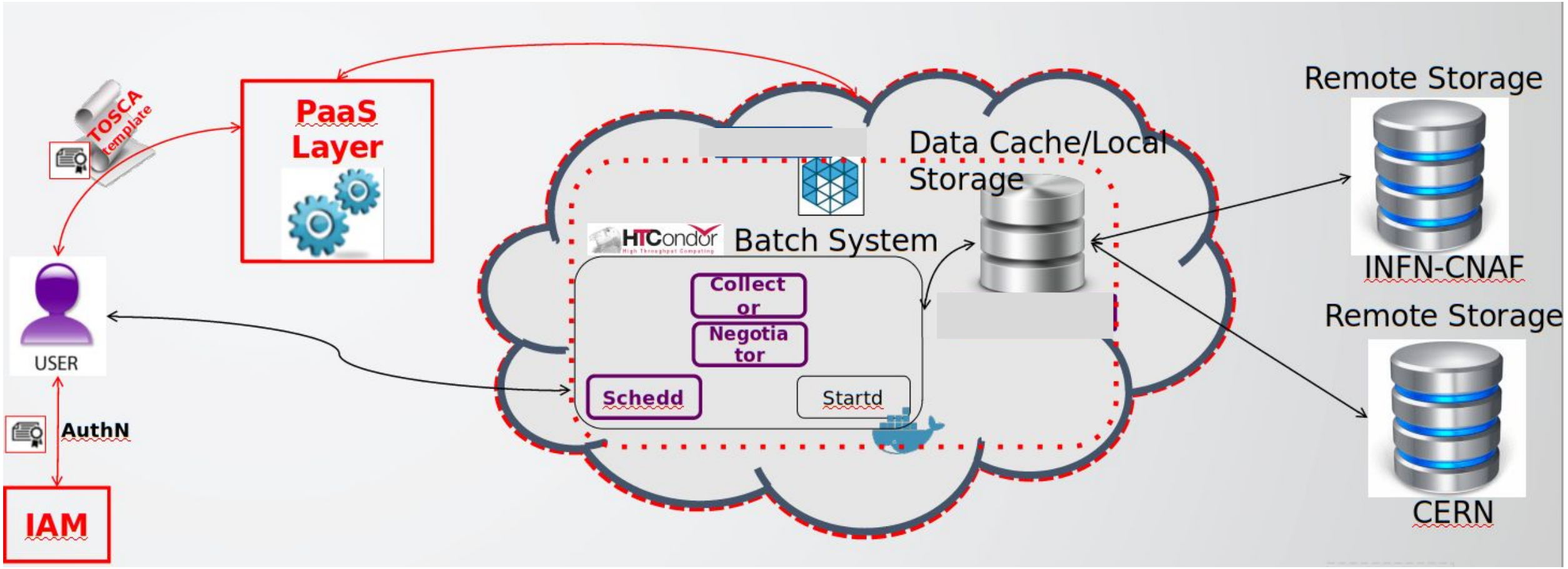
Putting everything together…

We realized that we had the technology to start filling the gaps…

As anticipated: we decided to try to use DODAS Thematic Service originally implemented for CMS and adapt to AMS needs

- the underlying technology (the building blocks) has been originally developed to be experiment independent
- we first performed a requirement analysis (see later)
- Toward the end of 2017 we started prototyping the integration as feasibility study

The initial schema



Requirement analysis

Data Input:

Read from eosams or eospublic@CERN and from gpfs@CNAF

- the “official” AMS files
- the ancillary files

Compute

Need a Batch System. No special requirements here

Working Station/User interface

Need a place where the user keeps his working directories (As it is now for Italian users at CNAF)

Data Output

Write to eospublic / eosams@CERN or to gpfs@CNAF via XRootD

Software

Read the “static” libraries needed by the executable, common to all the users

Read the “user” libraries needed by the executable

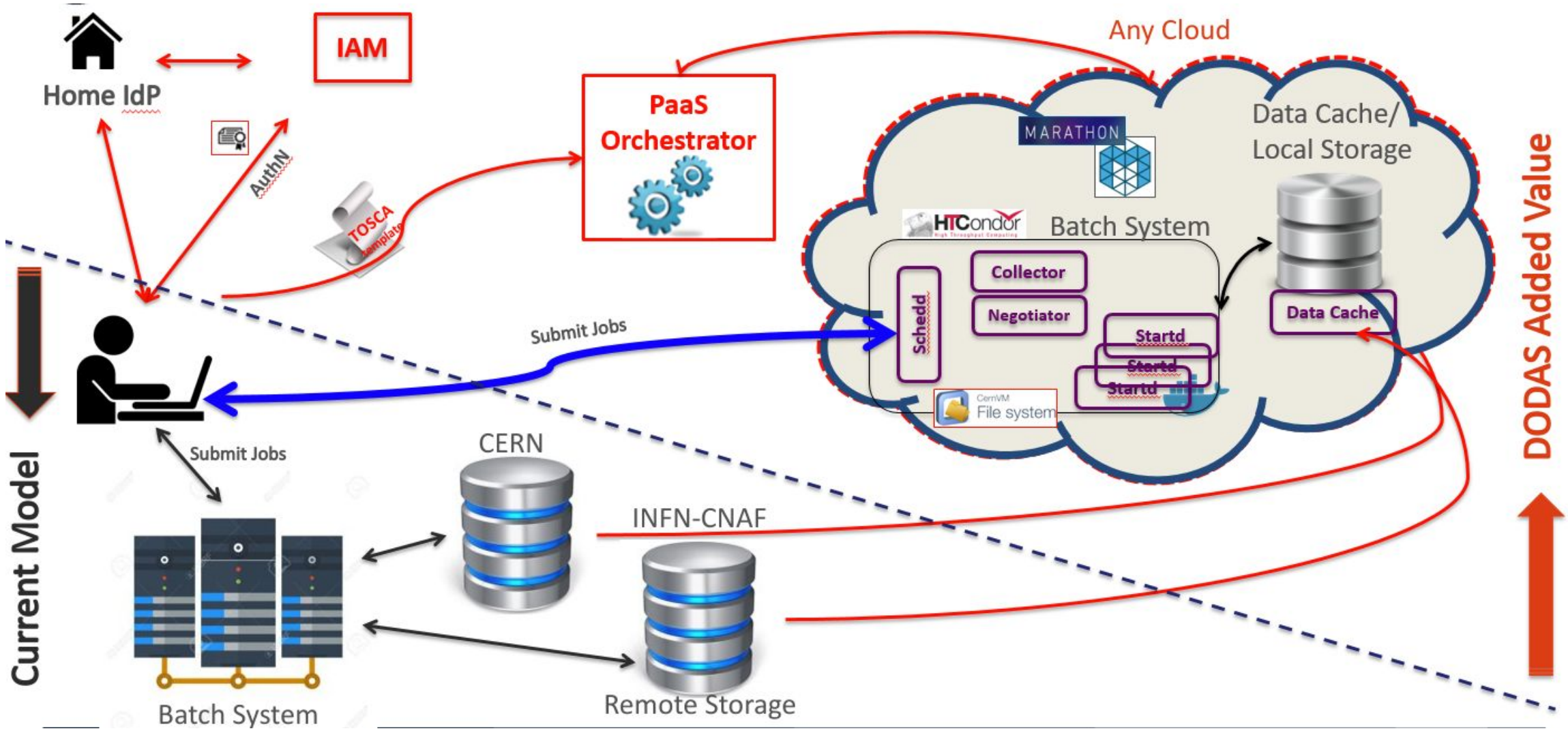
Technology mapping

- Data:
  - Data I/O is performed using XRootD. This technology enable, by construction, the possibility to implement
    federation, a key to use multiple data sources
    Data caching, a key to avoid latency issues while reading data from remote location
- Batch System:
  - HTCondor is the most suitable solution (many facilities including CNAF migrating, AMS already uses HTCondor at CERN etc)
    - moreover, we consider this a good choice also because offer many handles to implement federation.
    - we used it as overlay technology (see later)

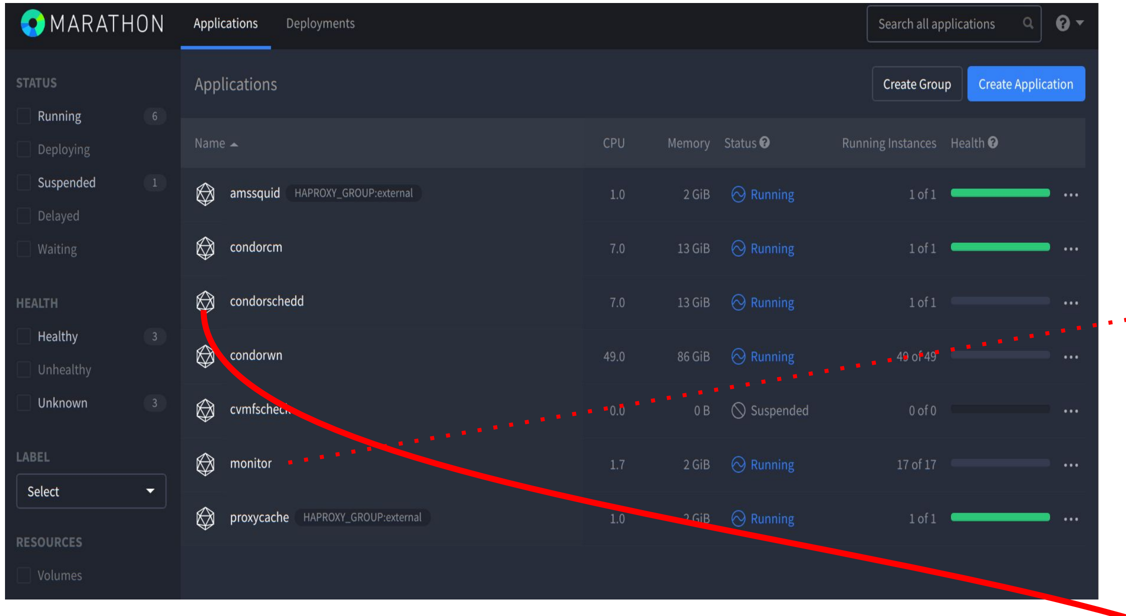
Technology mapping (cont)

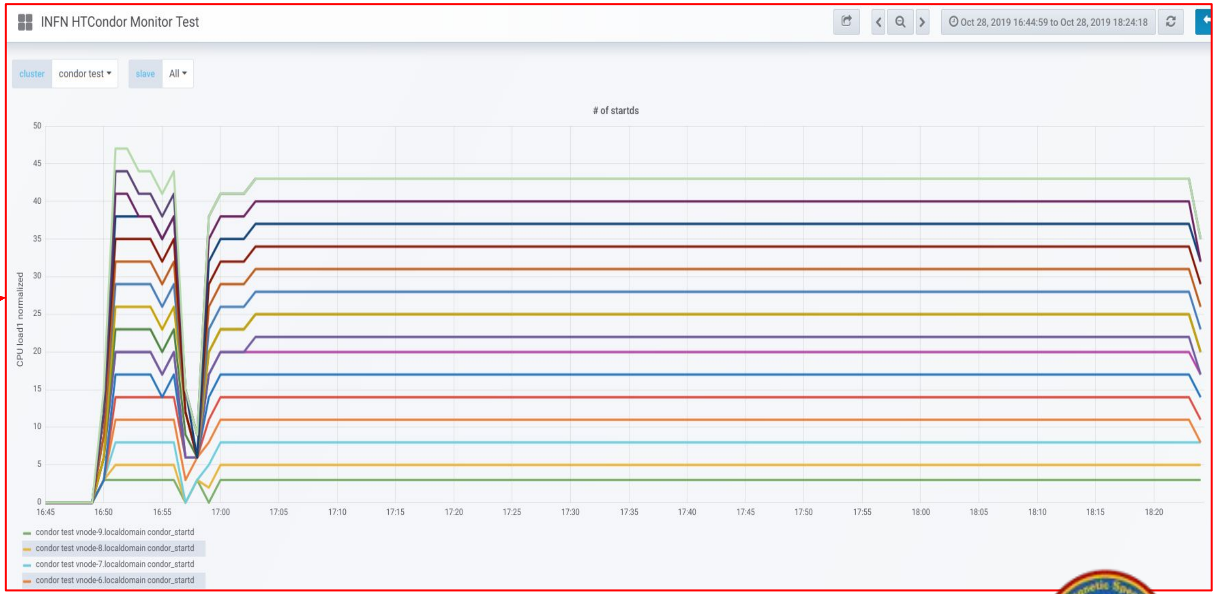
- Software
  - Several options have been evaluated and we concluded to proceed with CVMFS as the most suitable technical solution
    - experiment software already managed through CVMFS
    - user/group “custom” libraries stable enough (changes are >> hourly)
    - however this requires a piece more in the infrastructure and a action more from user perspectives
- Working Station:
  - we decided to rely on CNAF for the early integration
  - No needs for third solution although we could easily implement a automated solution to create a dedicated UI

The first end-to-end integration



And looking under the hood





@ CNAF

openstack

```
CONTAINER ID        IMAGE                   COMMAND                  CREATED             STATUS
       PORTS
                                                                      NAMES
77193b37a7e2        dodasts/ams:v1.0.0.rc1   "/usr/bin/tini -- /u…"   2 weeks ago         Up 2 weeks
                                                                      mesos-4e3d9c34-bbee-4c41-8964-37
b65b0a1951
9c9d621a58dc        progrium/consul:latest   "/bin/start -adverti…"   2 weeks ago         Up 2 weeks
       0.0.0.0:8300-8302->8300-8302/tcp, 0.0.0.0:8400->8400/tcp, 0.0.0.0:8500->8500/tcp, 0.0.0.0:8301-8
302->8301-8302/udp, 172.0.17.1:8600->53/tcp, 172.0.17.1:8600->53/udp   consul
root@scheddpublic:~#
```

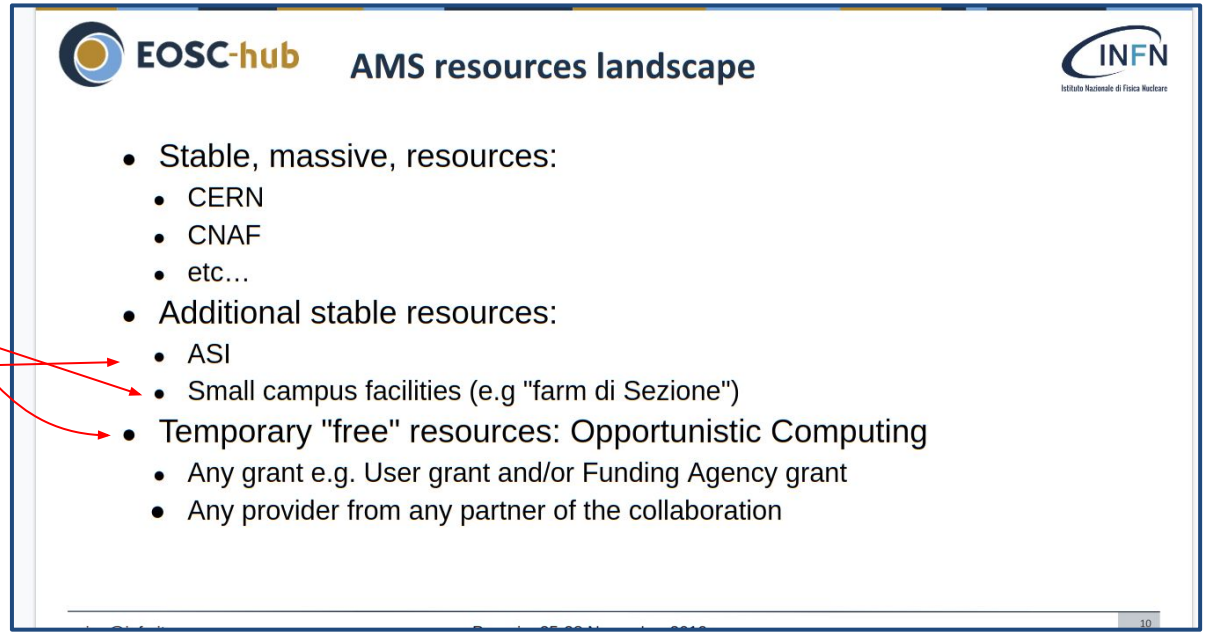
```
[root@scheddpublic ~]# condor_q

-- Schedd: scheddpublic.localdomain : <193.204.89.85:31206> @ 10/28/19 15:56:18
OWNER BATCH_NAME      SUBMITTED   DONE   RUN    IDLE   HOLD  TOTAL JOB_IDS

Total for query: 0 jobs; 0 completed, 0 removed, 0 idle, 0 running, 0 held, 0 suspended
Total for spiga: 0 jobs; 0 completed, 0 removed, 0 idle, 0 running, 0 held, 0 suspended
Total for all users: 411 jobs; 411 completed, 0 removed, 0 idle, 0 running, 0 held, 0 suspended
```

A first scale test during 2018

- A first integration finished during August
  - **Remote submission of analysis jobs from CNAF AMS-User Interfaces**: condor submit –pool XXX
    - Very same UI used to submit to LOCAL LSF
  - Input Data read from AMS **EOS @ CERN**
  - **Jobs run@TSystem** (Open Telekom Cloud)
  - **Produced output copied** on a temporary storage@TSystem (XRootD)
    - Third party copy from XRootD@OTC **to srm://storm-fe-ams.cr.cnaf.infn.it:8444**
- A scale test during Aug.-Sept. performed with real workflow (Matteo Duranti
  - 30+TB of produced data (ntuple production)
  - About 500K jobs

Implemented scenarios

- Batch system on demand
  - opportunistic resources
  - exploiting our own (INFN) resources
    - ReCaS@Bari and Cloud@CNAF
- Exploitation of ASI resources
  - see later for details
- Moreover we worked on federation of distributed resources through overlay HTCondor
  - ReCaS@Bari and OpenStack@Perugia

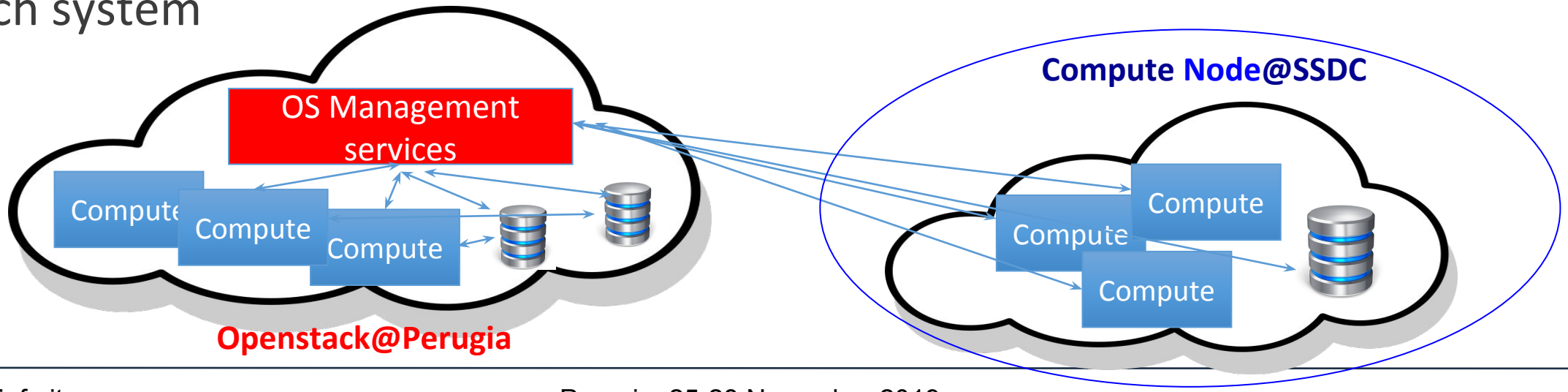


Computing resources@ASI

At the Space Scientific Data Center (SSDC) of the Italian Space Agency (ASI) we have an AMS farm: 384 cores, 90TB

- there is no dedicated support to AMS, researchers take care of sysadmin roles
- access to resources only via local batch
  - data must be copied manually etc.

AMS wanted to minimise the computing effort… so the idea was to extend resources management via Openstack (as a zone) and use DODAS to implement a batch system

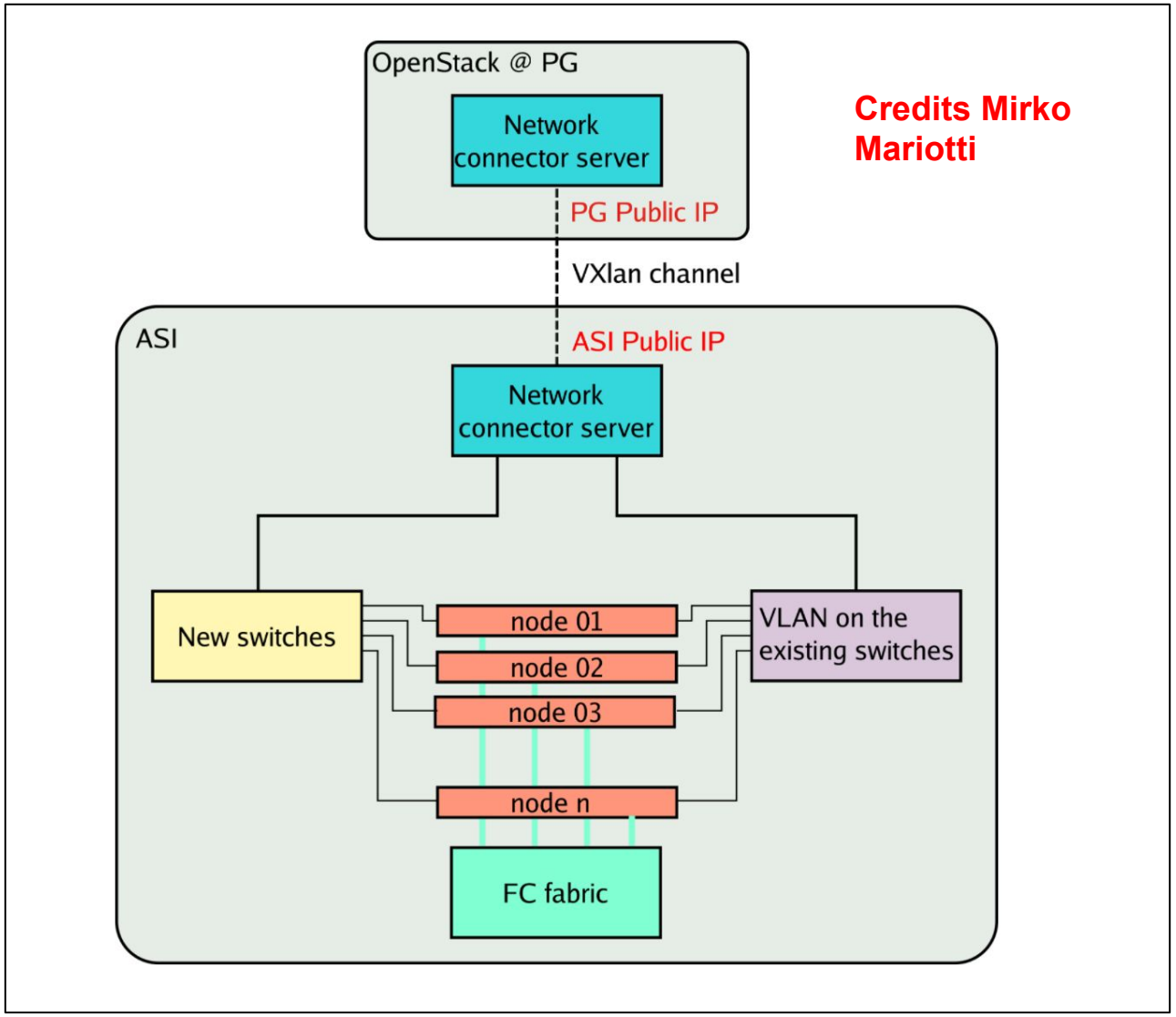


The underlying implementation

Il modello proposto si basa fondamentalmente su un overlay network per Il **management dei servizi di OpenStack** geograficamente distrubuiti

- **Sicurezza:**
  - Comunicazione remota dei servizi OS. Crittografia forte
  - Applicazioni utente finale: tutto può rimanere come oggi. Stesse policy autorizzative
    - Ovviamente il setup proposto offre altri gradi di libertà
- **Bandwidth:**
  - L'overhead rispetto al setup attuale è minimo (solo traffico di funzionamento OS)

Nota: L'obiettivo del proposal è che il traffico sul canale INFN-PG □□SSDC serva **solamente** al management di OpenStack

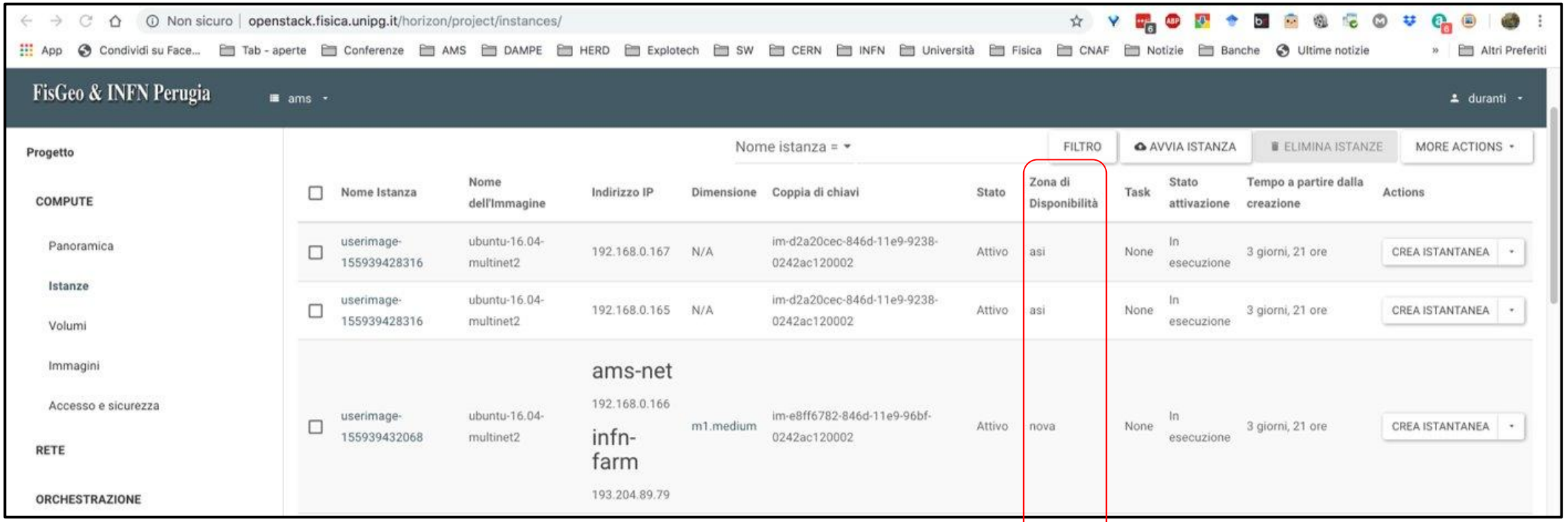


Credits Mirko Mariotti

And from users perspectives

Everything is transparent. Just see more worker nodes

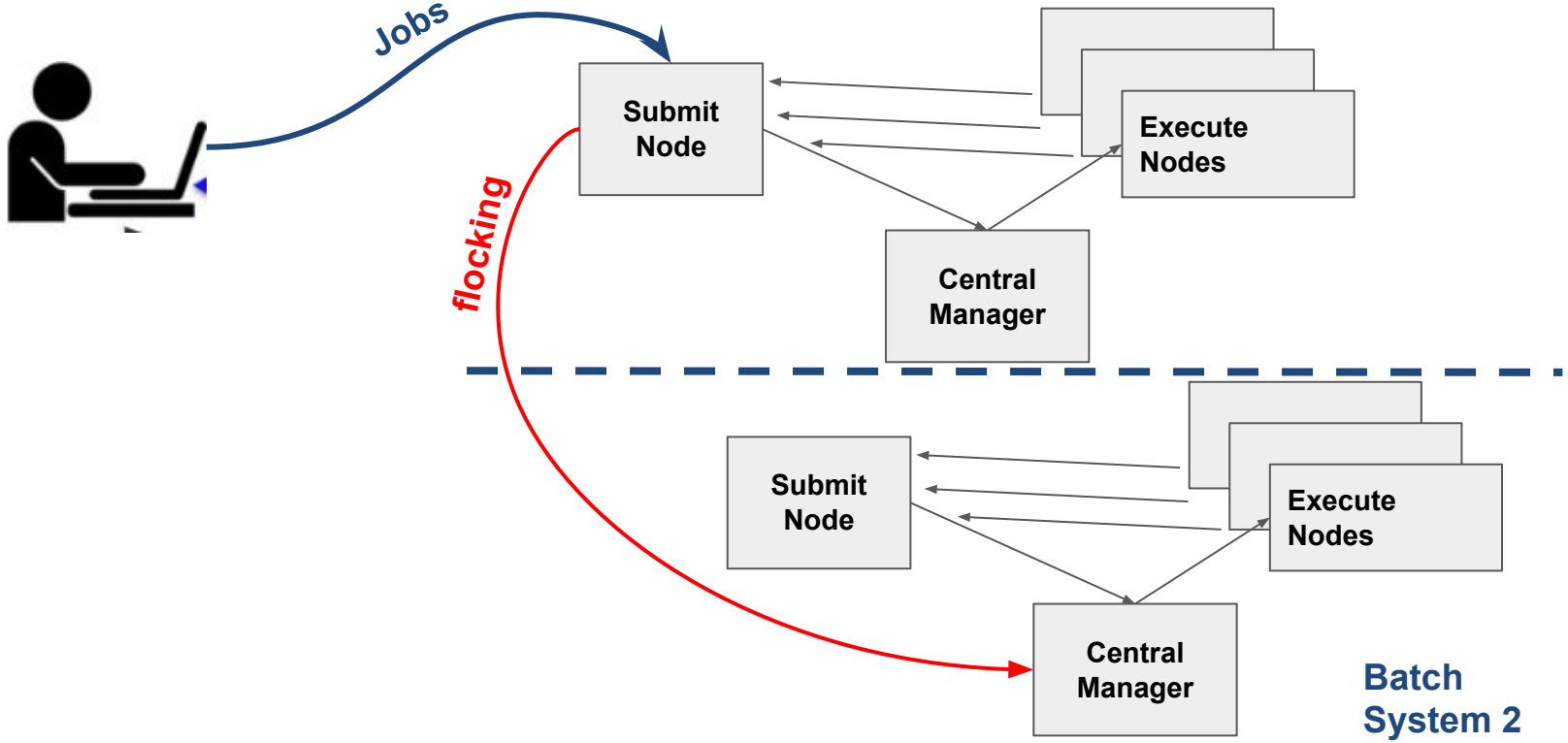
- We deployed (automatically) a cluster where HTCondor services run on Openstack @PG and WorkerNodes (startd) are in ASI



Federation through HTCondor Flocking

AKA a native HTCondor federation solution

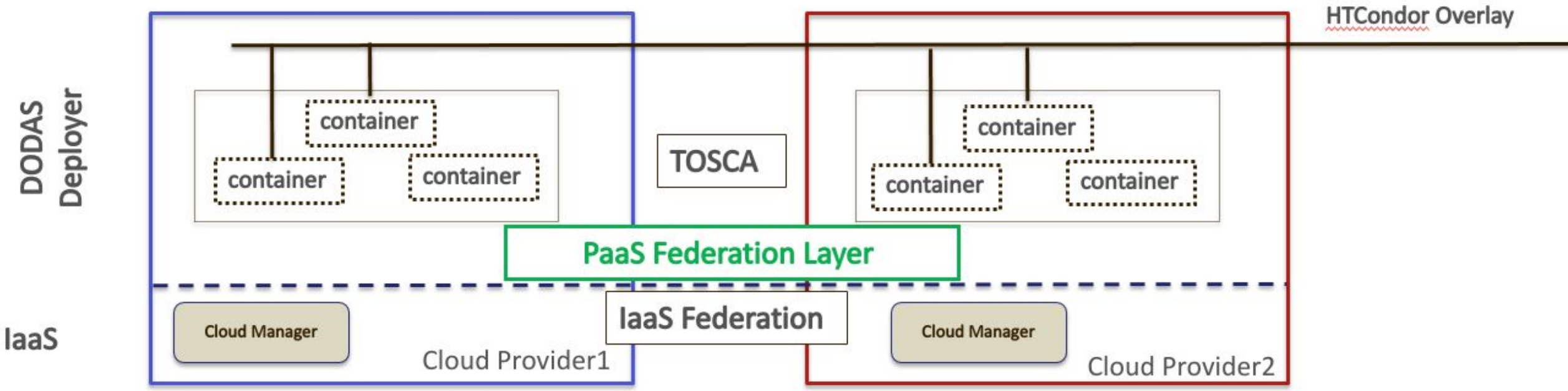
- easy to implement from the HTCondor perspectives
  - few lines of config file (flock from, flock to)
  - plus AuthN/Z
- It is transparent from the user point of view
  - just keep submitting jobs as before



From the infrastructure point of view

Openstack cloud provider in Bari and In Perugia

PaaS federation layer allow to build container based HTCondor overylay



Authentication and Authorization

Remote submission and/or federation of distributed pools requires authentication

So far we rely GSI which impy X509 certificates

- we rely on INDIGO-IAM as a building block in this context
  - this start from a OIDC Token and has capability to deliver also X509

This is a key point because:

- allows integration with legacy system
- does not require any extra knowledge to the user
- it provide the status of art, which means we are ready for integrating services and systems supporting OIDC (Json Web Token).
  - we will see in practice in the next days. We'll discuss a bit in detail on Thursday

In Summary, what happened so far...

- Resources exploited (rough extimation):
  - 700k jobs (2-3 hours each 1000 cores * 2 months ≈ 15-20% of resources available at CNAF for AMS)
  - 30TB generated data (copied to CNAF)

- Used Infrastructures:
  - HelixNebula Science Cloud (TSystem)
  - Google Cloud
  - Cloud@CNAF
  - ReCaS@Bari
  - OpenStack@PG
  - OpenStack@PG-ASI (not yet at scale)

Gaps to Fill… or future work

The main gap (this might be a bit INFN biased) is to improve the federation strategy

- Both in term of policies and also from an architectural point of view
  - global view of harvested resources
- Monitoring and accounting still to be improved
- better documentation….

The main objective for the future is to consolidate the current work and possibly to extend it to future experiments and collaboration

- this is under discussion (very preliminary)

Thank you for your attention!

Questions?

Contact

eosc-hub.eu @EOSC_eu