

XENONnT

Analysis software:

CernVM Users Workshop 2022



Joran Angevaare (Nikhef & GRAPPA)

j.angevaare@nikhef.nl

13 September 2022



XENON

Nikhef

GRAPPA



GRavitation AstroParticle Physics Amsterdam

XENON Family:



Columbia



KIT



Nikhef



Muenster



Stockholm



Mainz



MPIK, Heidelberg



Freiburg



Zurich



Chicago



UCSD



Rice



Purdue



Subatech



Coimbra



LPNHE



Torino



Bologna



L'Aquila



LNGS



Weizmann



Tsinghua



Tokyo



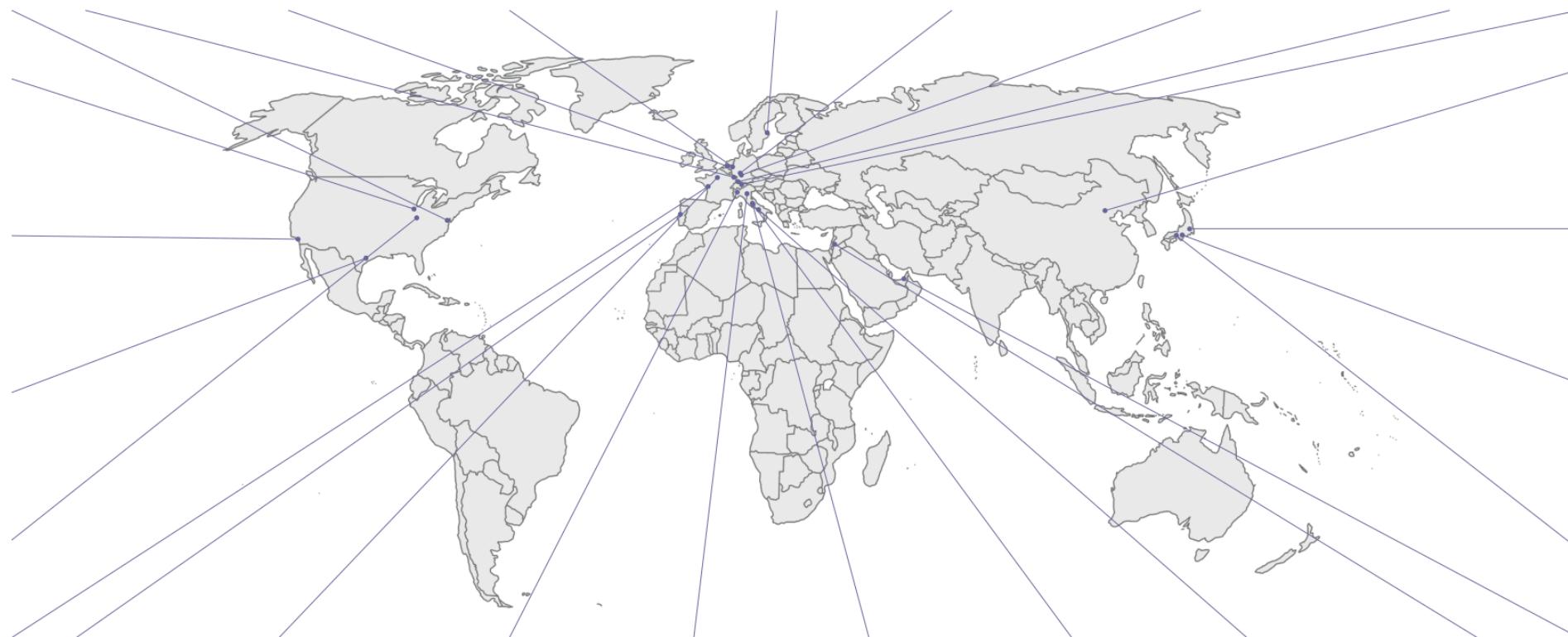
Nagoya



Kobe



NYUAD



XENON Family:



UC San Diego

UCSD



Subatech

Coimbra

LPNHE

Torino

Bologna

L'Aquila

LNGS

Napoli

Weizmann

NYUAD



Zurich



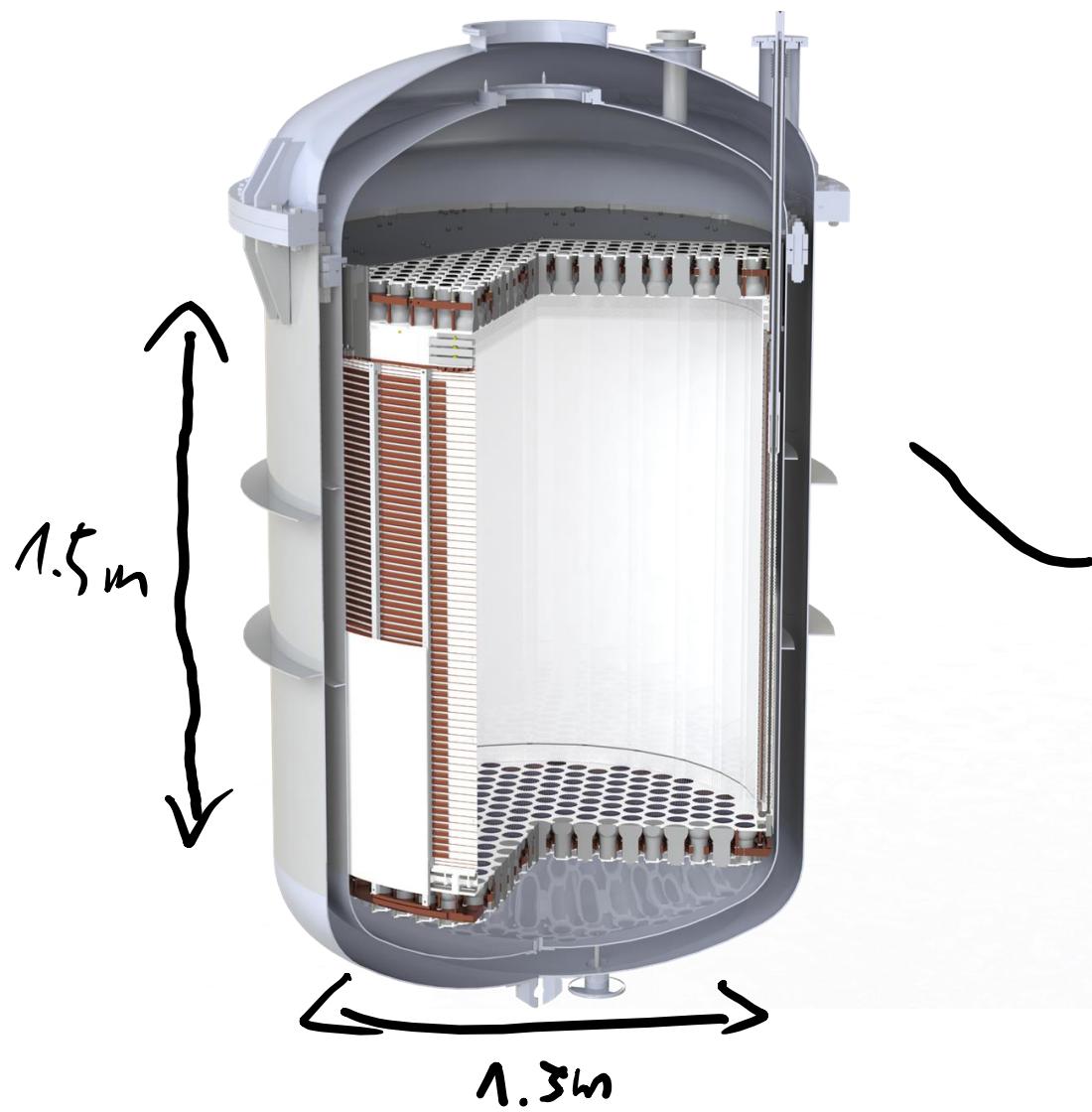
Tsinghua



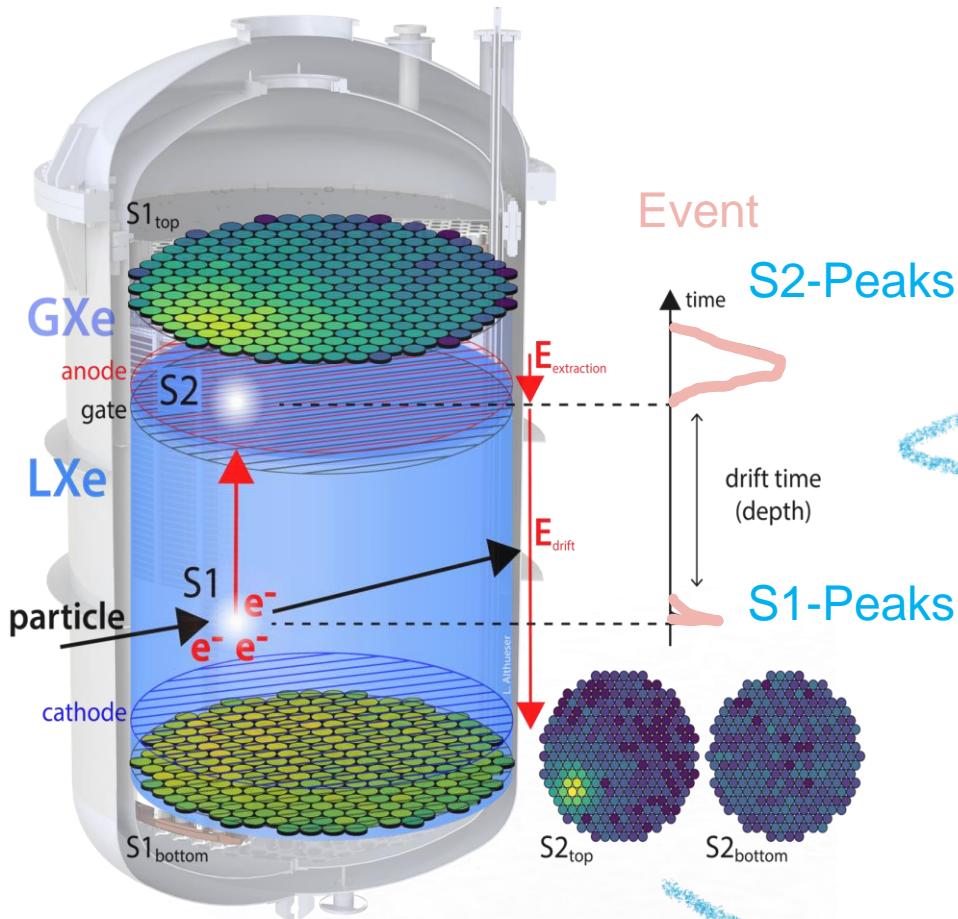
Tokyo



The XENONnT Experiment:



The XENONnT Experiment:



Event
S2-Peaks

S1-Peaks

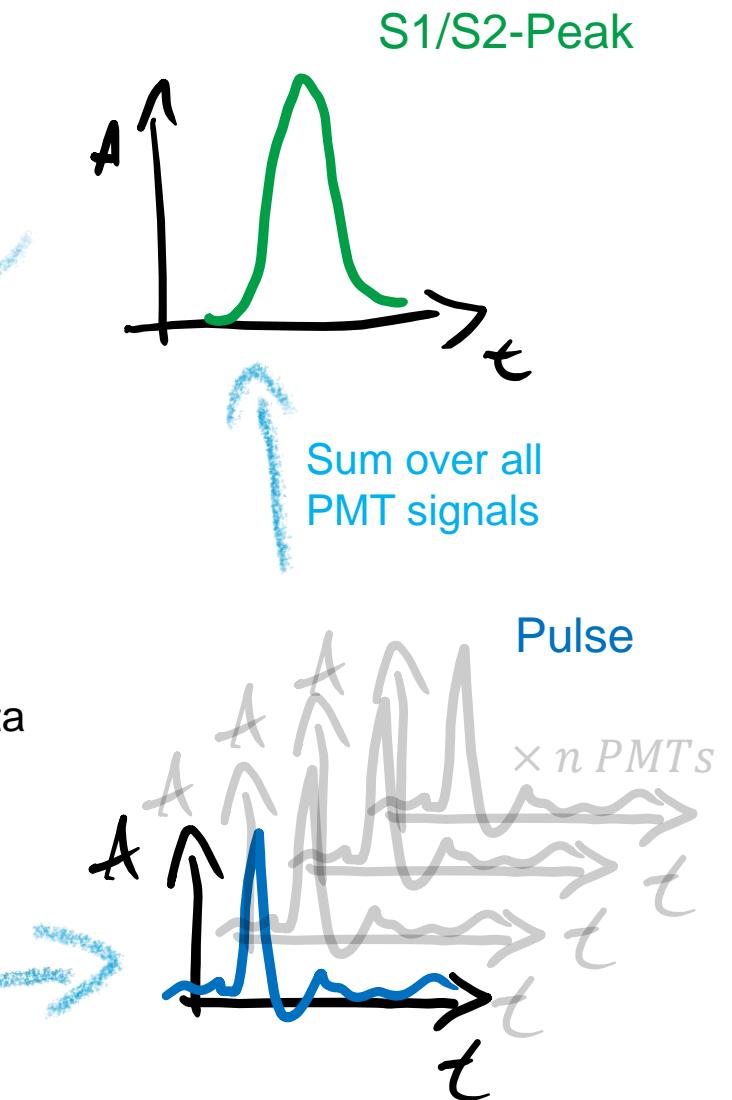
S2_{top}

S2_{bottom}

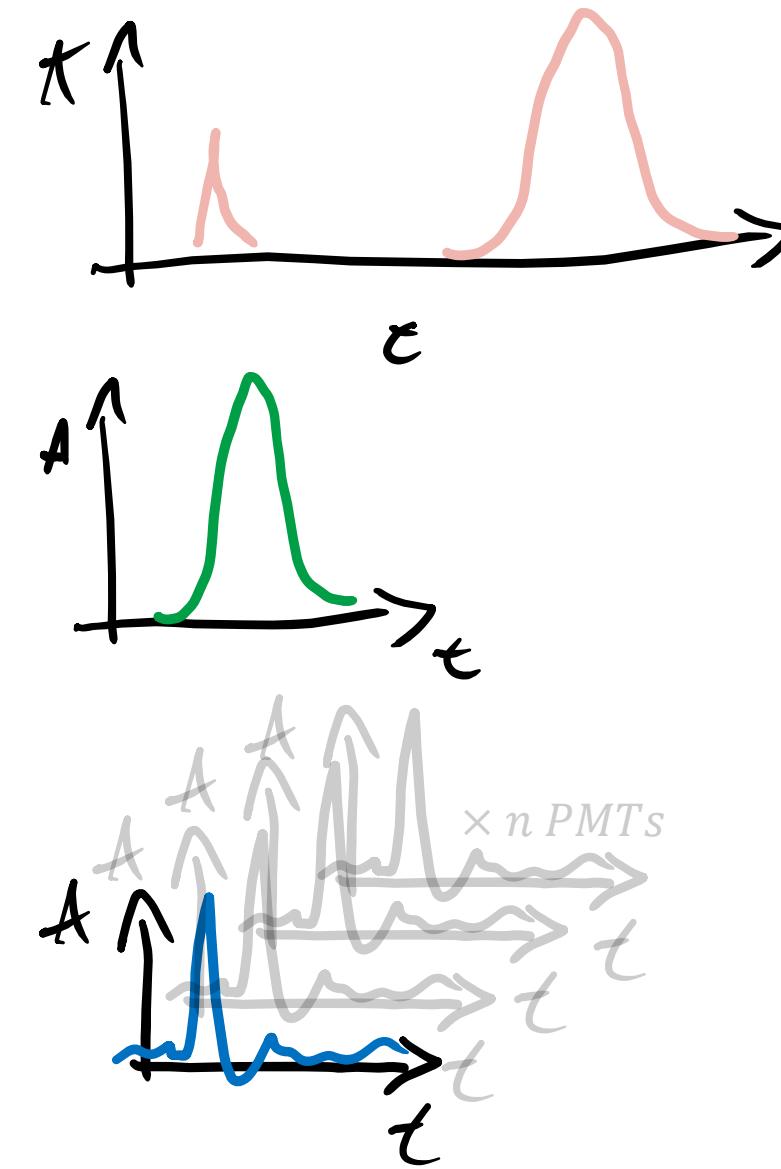
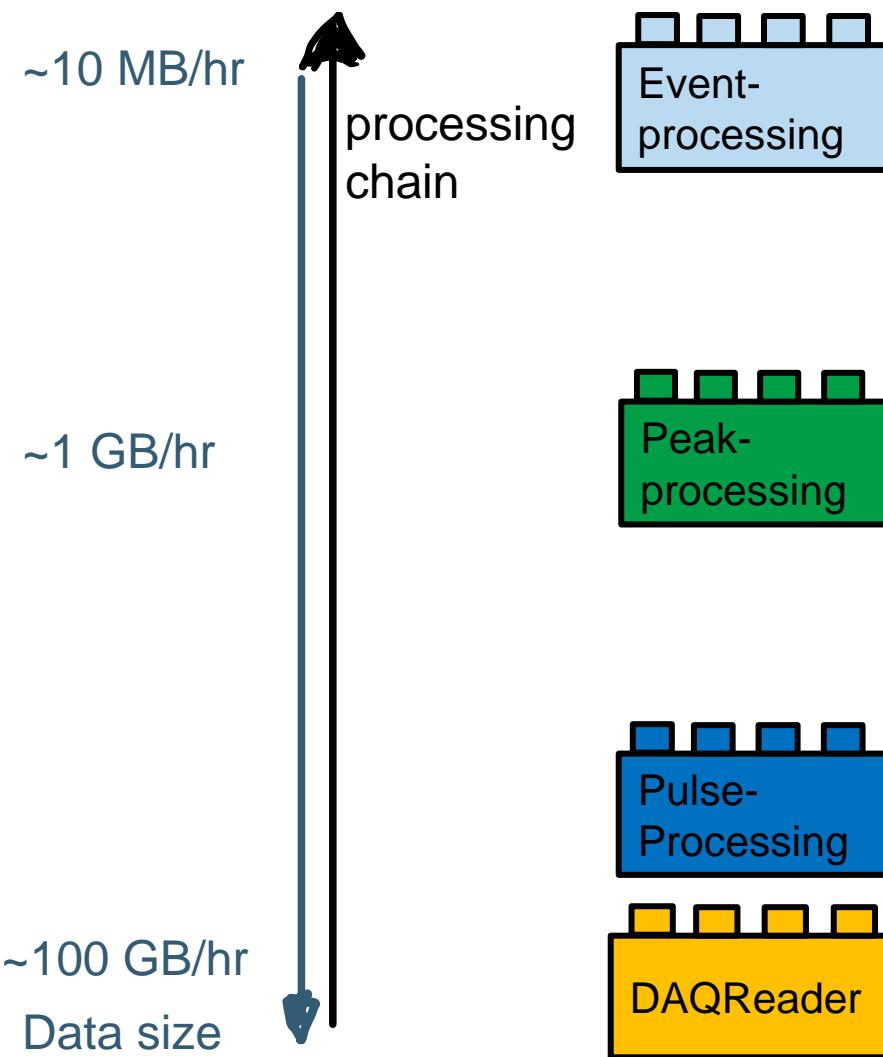
Selftriggered, 10 ns binned data
redax: 50 MB/s
LZ: 98.3 MB/s⁽¹⁾ (triggered)
Atlas: 340 MB/s⁽²⁾ (triggered)

(1) LZ technical design Report, LZ Collaboration
[arxiv:1703.09144](https://arxiv.org/abs/1703.09144)

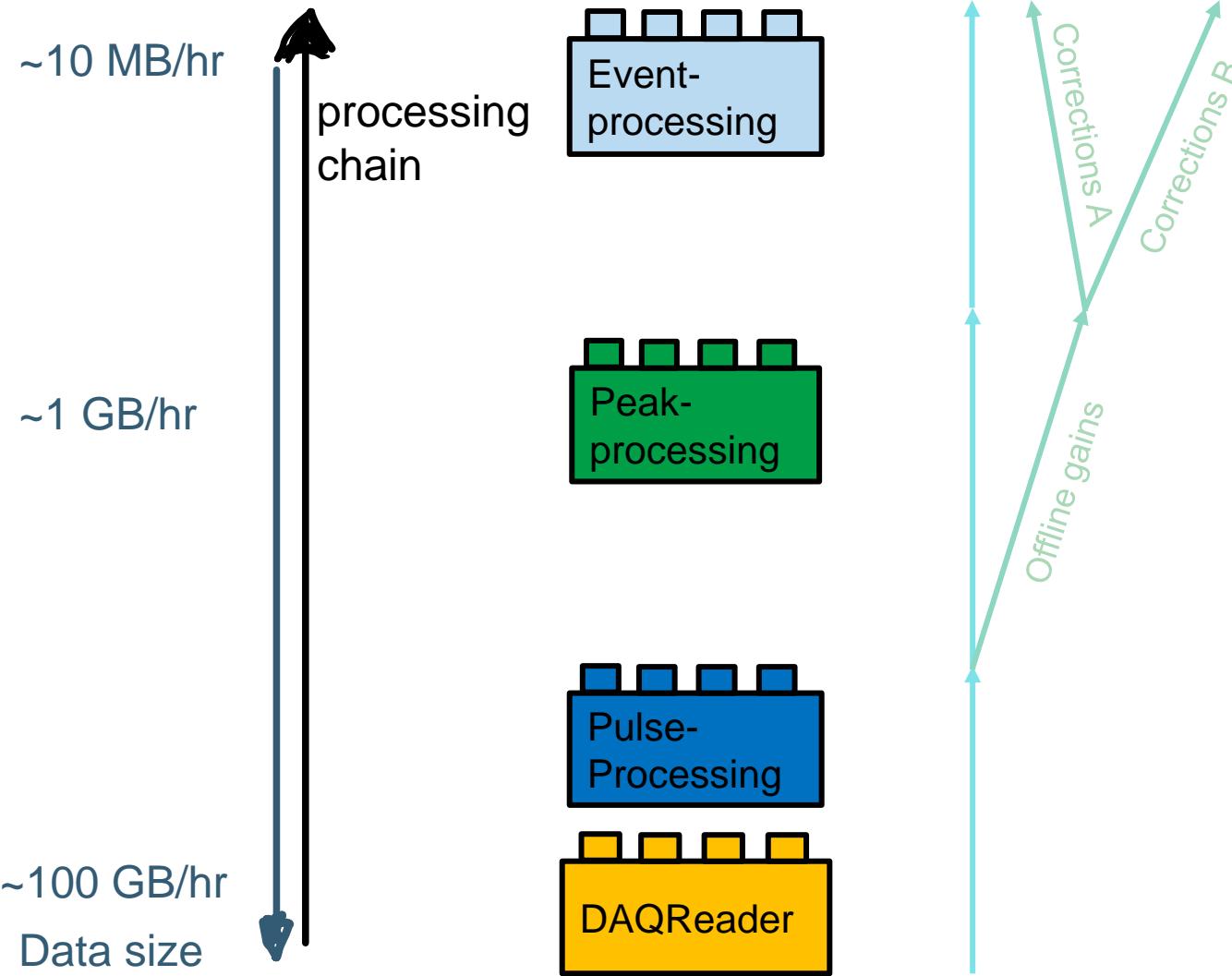
(2) ATLAS Fact Sheet, Atlas Collaboration



Processing software



Processing software



Changes in processing settings tracked per data-type

- Low level data never reprocessed
- High level data regularly processed for: PMT gains, corrections, etc.

Two types of configuration sets:

- **Online**; for fast monitoring and immediate feedback
- **Offline**; for science publications

XENON software

Data processing:

- Fully in Python



- Strax



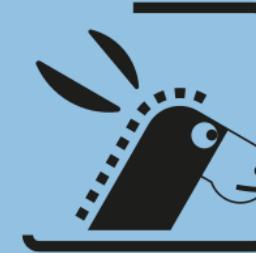
- Straxen



Data management:

- Fully in Python

- Rucio



- Admix



Simulation:

- Python + GEANT4



- WFSim



XENON

- mc (private)



XENON

XENON software

Data processing:

- Fully in Python
- [Strax](#)
- [Straxen](#)

Data management:

- Fully in Python
- [Rucio](#)
- [Admix](#)

Simulation:

- Python + GEANT4
- [WFSim](#)
- [mc \(private\)](#)

The base environment

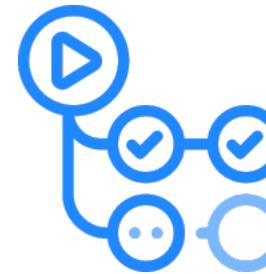


The Monte Carlo environment



Data processing:

- Split between:
 - general & optimized code ([Strax](#))
 - detector specific software ([Straxen](#))
- Continuous integration for testing:
 - [Github actions](#)
 - Many more
- The core code is public
- Analyses & detector conditions private



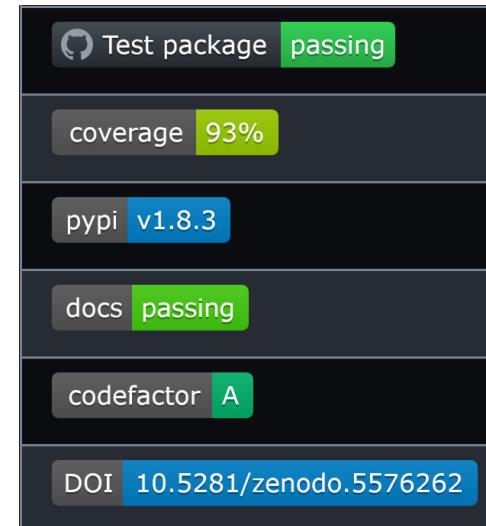
AxFoundation
Place holder

Overview Repositories 3 Projects Packages Teams 1 People

Popular repositories

strax Stream analysis for xenon TPCs Python 19 34	redax Forked from coderdj/redax Distributed, parallel high-throughput software for XENONnT with NoSQL https://github.com/axfoundation/redax C++ 3 6
---	---

<https://github.com/AxFoundation>



XENONnT Dark Matter experiment
Software for the XENONnT experiment
Gran Sasso, Italy https://en.wikipedia.org/wiki/XENON

Overview Repositories 59 Projects 3 Packages Teams 1 People 153

Popular repositories

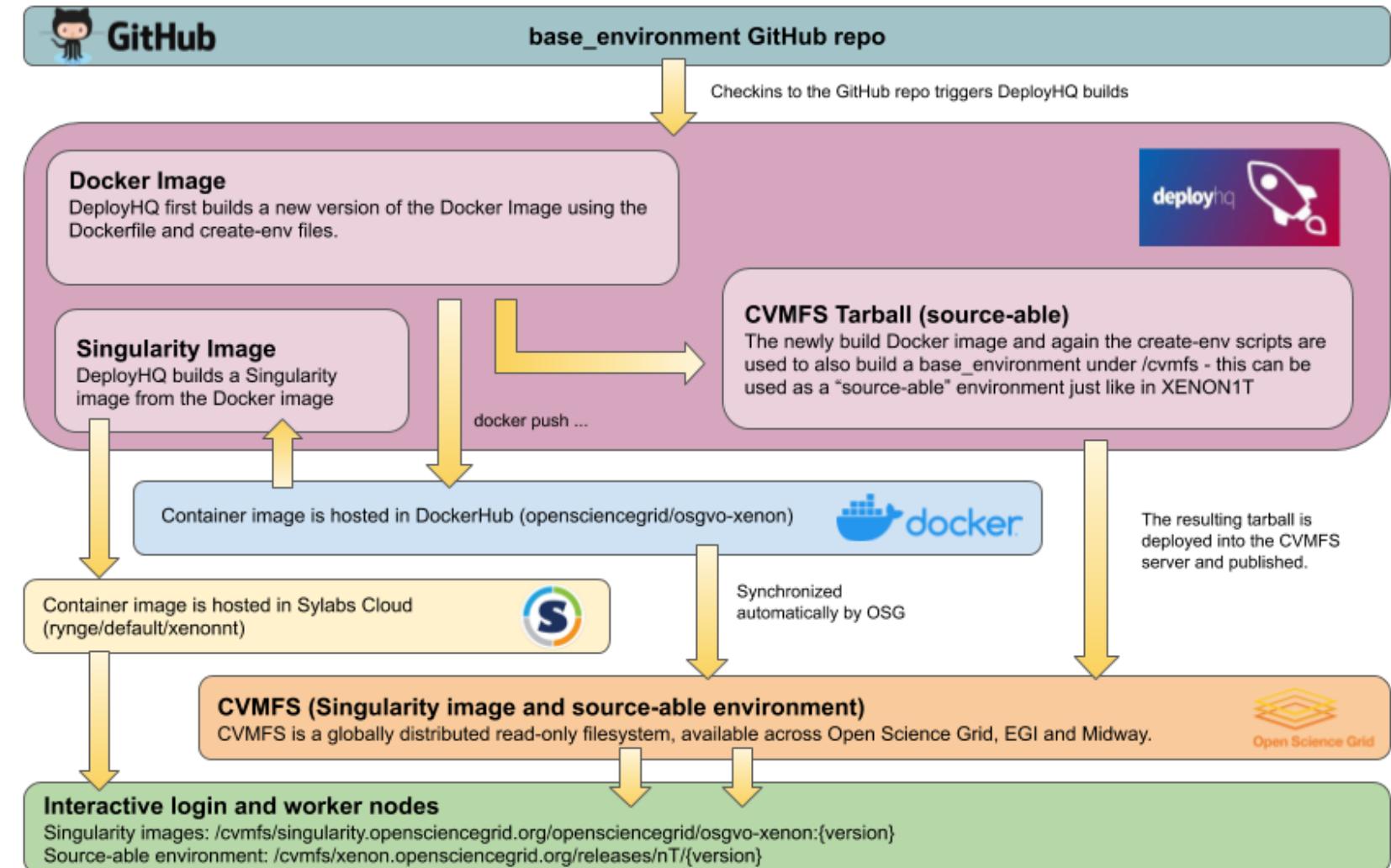
straxen Streaming analysis for XENON Python 11 30	WFSim Waveform simulator Python 7 7
---	---

<https://github.com/XENONnT>

The base environment



- Build docker image from github: [base environment](#)
- Auto builds & deploys
- Tracks configuration sets



Versioned containers

- Dockerized environment ~2x per month
- Distributed via CVMFS



```
[jorana@stbc-i1 ~]$ cd /cvmfs/xenon.opensciencegrid.org/releases/nT
[jorana@stbc-i1 nT]$ ls
2019.11      2020.11.25  2021.08.1   2021.10.5   2021.11.6   2022.01.4   2022.03.4   2022.06.3   stable
2020.02      2020.12.21  2021.08.2   2021.11.1   2021.12.1   2022.02.2   2022.03.5   2022.07.27  straxen_0-13-1
2020.03      2020.12.23  2021.10.1   2021.11.2   2021.12.2   2022.02.3   2022.04.1   2022.09.1   straxen_v100
2020.04      2021.01.06  2021.10.2   2021.11.3   2021.12.3   2022.02.4   2022.04.2   development
2020.05      2021.04.18  2021.10.3   2021.11.4   2022.01.2   2022.03.1   2022.04.3   gpu
2020.11.06   2021.05.04  2021.10.4   2021.11.5   2022.01.3   2022.03.3   2022.06.2   py38
[jorana@stbc-i1 nT]$
```

XENON (analyses) sites

DAQ
@
LNGS (Italy)



Fully live processing

Live monitoring

Triggerless data
digitization and long
term storage by
heavy compression

Data
manager
@
LNGS
(Italy)

Upload to
Rucio

Midway / Dali
@
UChicago



Main analysis site
Full copy of
processed data

Fried rice
@
Rice University



GPUs

Private server

Jupiter lab

Mounts Midway

XENON (analyses) sites

DAQ
@
LNGS (Italy)



Fully live processing

Live monitoring

Triggerless data
digitization and long
term storage by
heavy compression

Data
manager
@
LNGS
(Italy)

Upload to
Rucio

Midway / Dali
@
UChicago



Main analysis site
Full copy of
processed data



CVMFS

Fried rice
@
Rice University



GPUs
Private server
Jupiter lab
Mounts Midway



planned
CVMFS

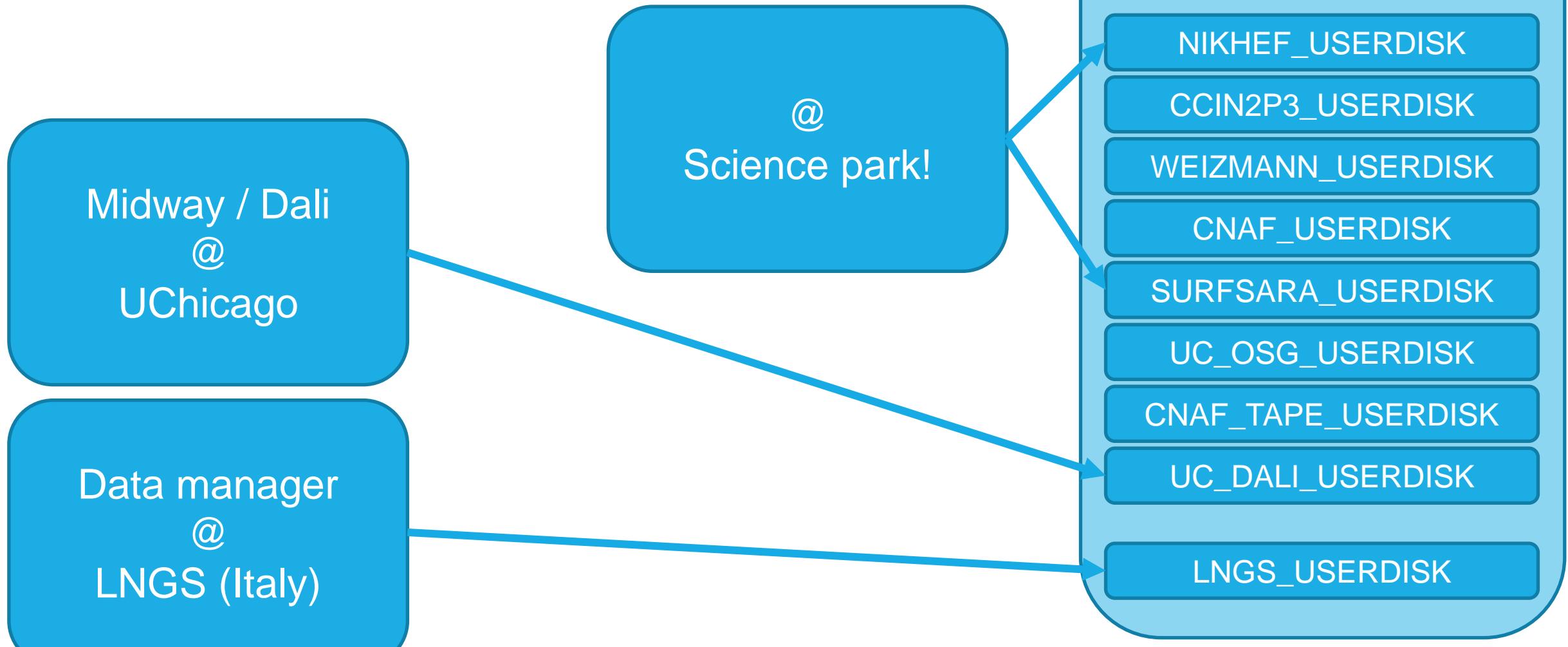
Distributed processing

- (raw) data distributed among Rucio storage sites



Distributed processing

- (raw) data distributed among Rucio storage sites



Distributed processing

- (raw) data distributed among Rucio storage sites
- Reprocessing for offline software configuration:
 - Grid processing! Using [outsource](#)
 - If high level data, even one cluster (midway) sufficient for reprocessing ([reprox](#))

Rucio



NIKHEF_USERDISK

CCIN2P3_USERDISK

WEIZMANN_USERDISK

CNAF_USERDISK

SURFSARA_USERDISK

UC_OSG_USERDISK

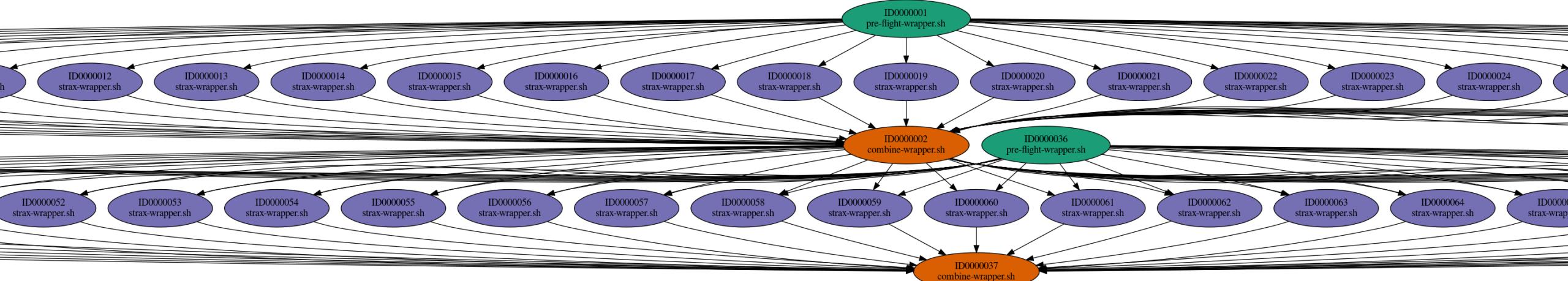
CNAF_TAPE_USERDISK

UC_DALI_USERDISK

LNGS_USERDISK

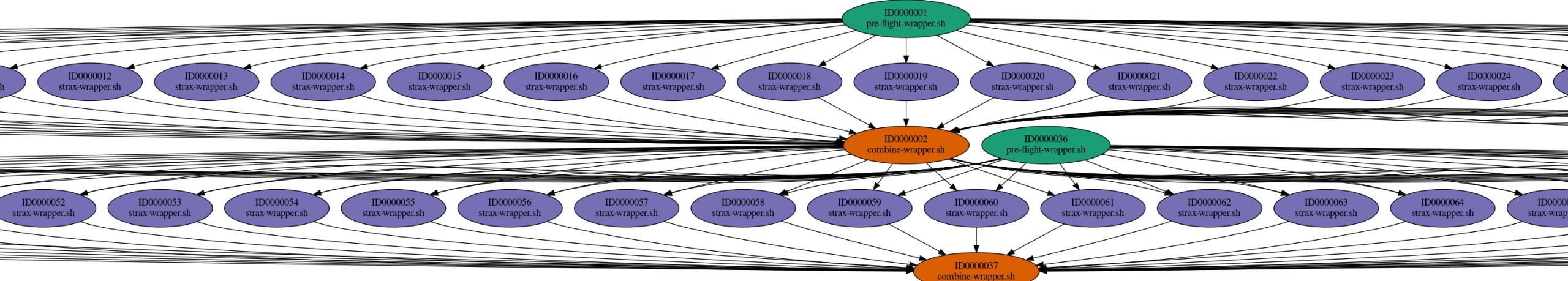
Distributed processing - outsource

- Wrapper around [Pegasus](#)
- Process data chunks $\sim \mathcal{O}(0.2 - 2)$ GB per job
- Leverages the Open Science Grid (OSG) and European Grid Infrastructure (EGI):
 - Thanks to e.g. [PDP@Nikhef](#) and SURFsara
- Example of workflow:



Distributed processing - outsource

- Ship configuration files
- Source environment: CVMFS
- Download chunk(s) with Rucio
- Stage input(s) for processing
- Process
- Merge outputs & upload
- Repeat?



Summary & outlook

- XENONnT digitizes triggerless data which is stored long term
 - Fast software, so data-size is computing's bottleneck
- Data handled and versioned per 'data-type'
- Processing performed in two configurations:
 - Online for live feedback & offline for science results
- Software is containerized in docker images:
 - Distributed via CVMFS to OSG, EGI and analysis sites
- Reprocessing on grid via outsource (Pegasus workflow)
- Future:
 - GPU containers
 - Easy ports to Jupyter hubs

Back-up