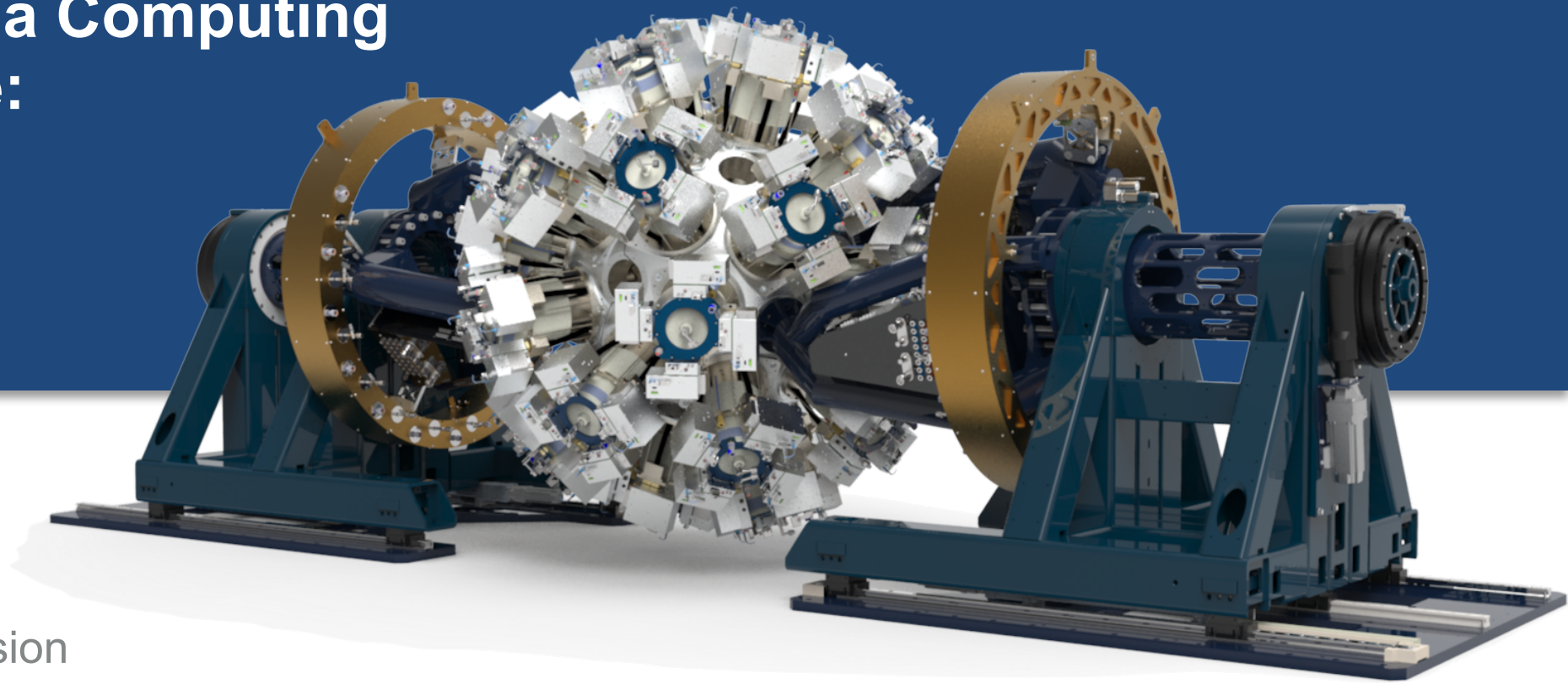
GRETA/Deleria Computing Infrastructure: Streaming



Mario Cromaz

Nuclear Science Division

Lawrence Berkeley National Laboratory

BERKELEY LAB

Bringing Science Solutions to the World

SRO XIV - Sept. 29, 2026

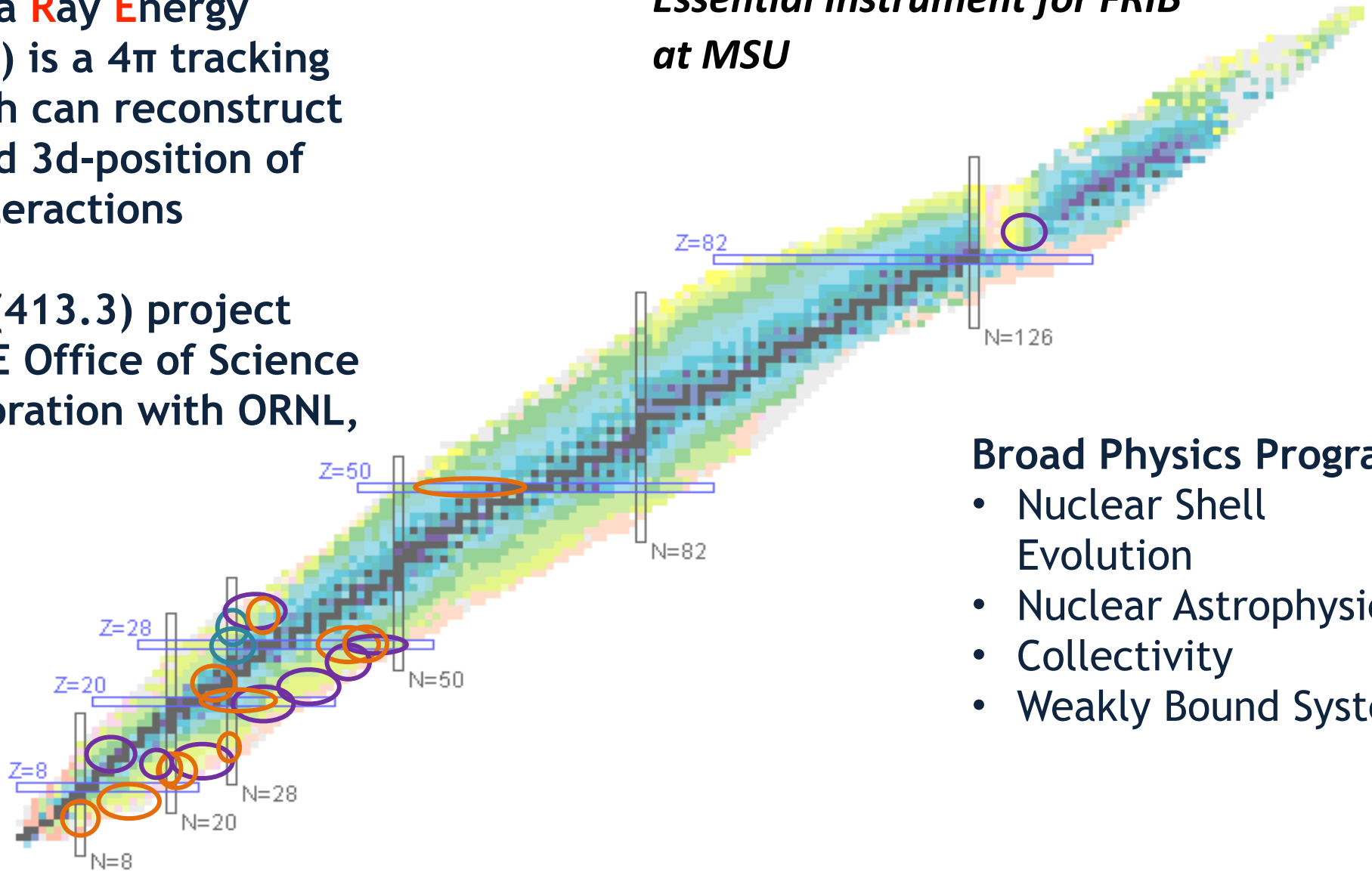
U.S. DEPARTMENT OF ENERGY | Office of Science

GRETA Science Program

GRETA (Gamma Ray Energy Tracking Array) is a 4π tracking detector which can reconstruct the energy and 3d-position of gamma-ray interactions

LBNL-led MIE (413.3) project funded by DOE Office of Science (NP) in collaboration with ORNL, ANL, FRIB

Essential Instrument for FRIB at MSU

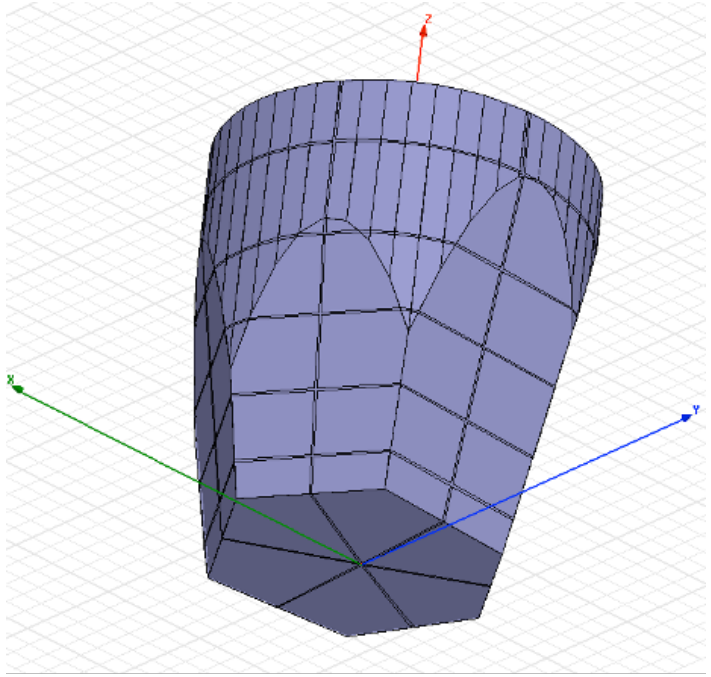


Broad Physics Program:

- Nuclear Shell Evolution
- Nuclear Astrophysics
- Collectivity
- Weakly Bound Systems

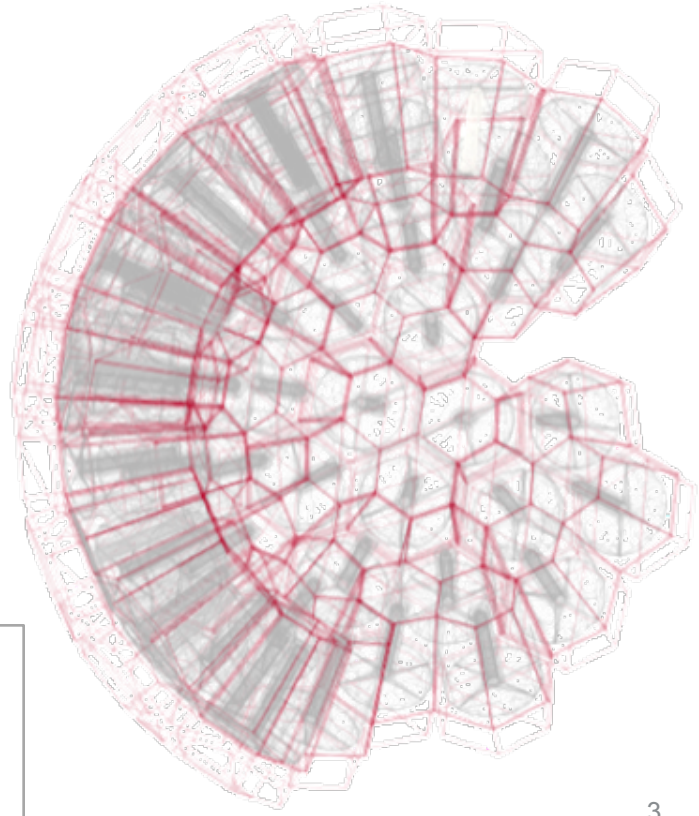
The Detector Elements - HPGe Crystals

- The detector material is a high purity Ge (HPGe) cooled to LN temperatures
- Large HPGe crystal (9cm long, 8cm diameter)
- Each crystal segmented - 6 longitudinal and 6 azimuthal segments
  - Produces 36 voxels
- Charge deposited in a voxel is collected on its corresponding segment pad



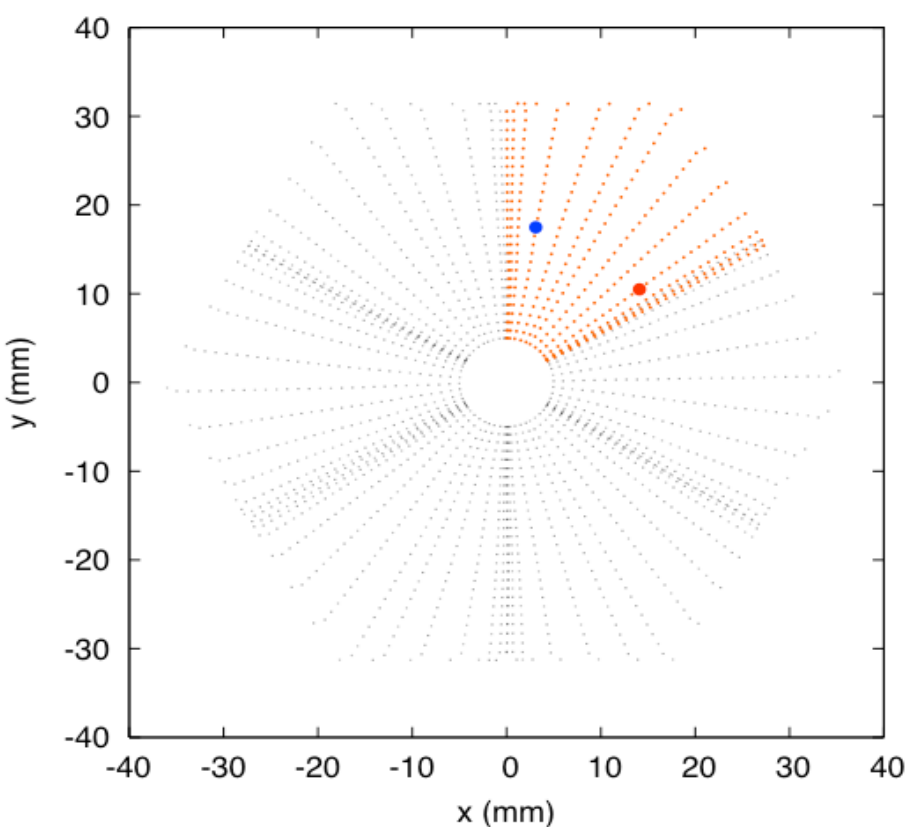
Schematic diagram of a hexagonal 36-way segmented HPGe detector

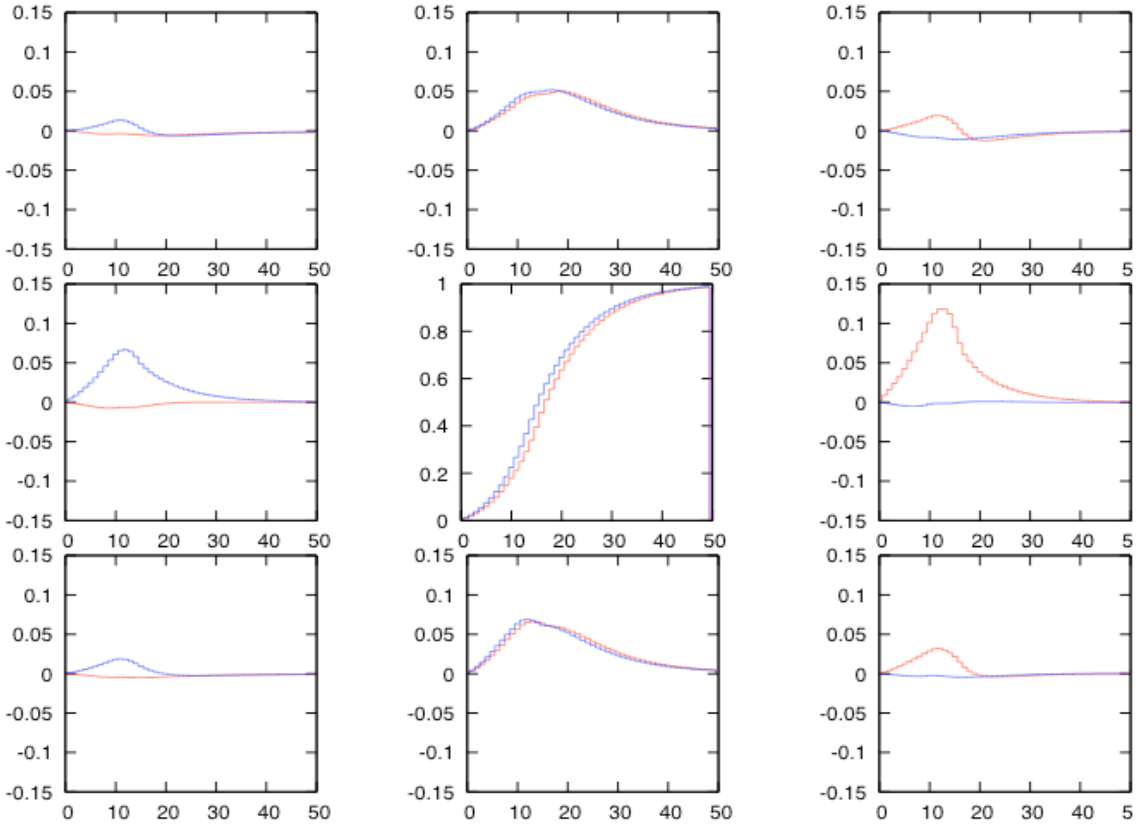
Close pack 120 crystals in a sphere around the target position to maximize efficiency



Tiling a sphere with 120 crystals
→ 4320 “effective” detectors

Locating Where the Gamma-Ray Interacted

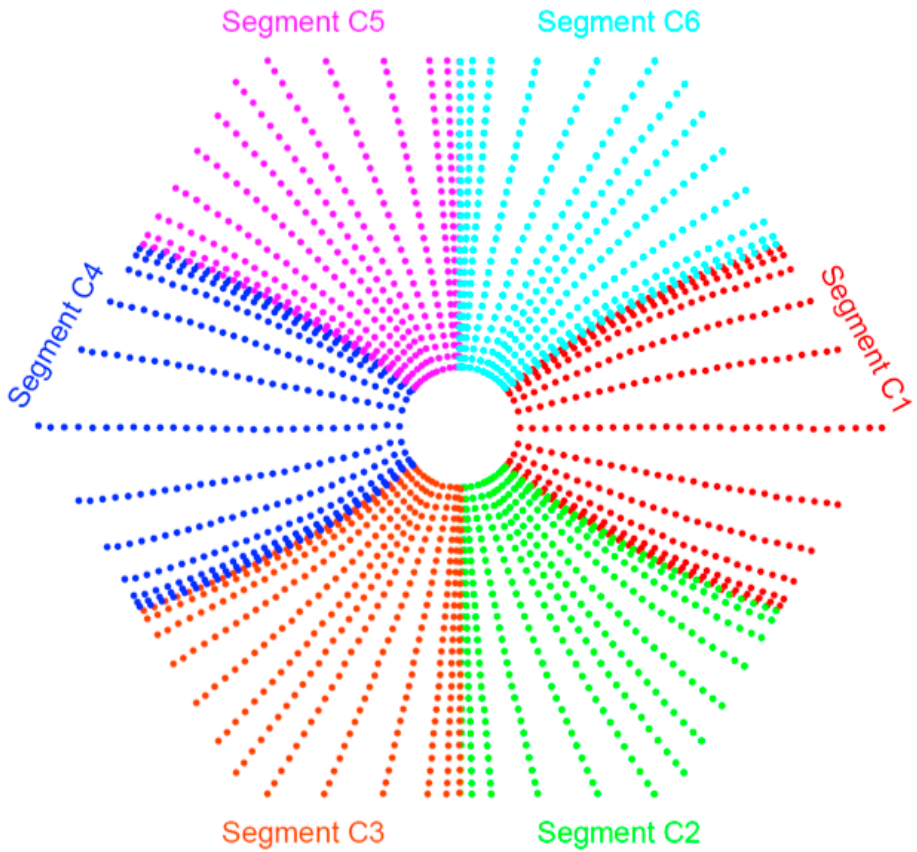




- Each point in the detector can be associated with a unique set of waveforms on the detectors pads
- Likely to be multiple such interactions per segment - experiment sees a superposition of these signals

The Inline Computation to Be Carried Out

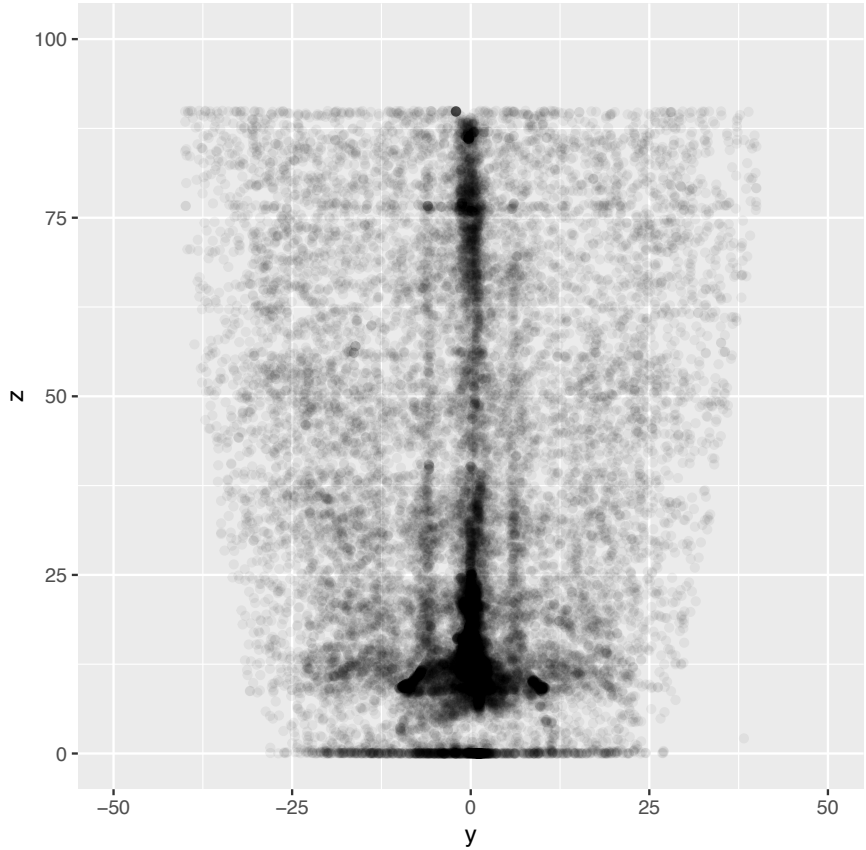
- Generate a grid and simulate charge collection on segment pads for unit charge on each grid point (~250,000 grid points)
- Search linear combinations of these signals to identify the positions which contributed to the sum signal ***(Signal Decomposition)***
- Current algorithm: adaptive grid search + non-linear least square fit
- ~5 ms / core / event



Grid point on equi-sensitivity grid in x-y plane
(D. Radford, K. Lagergren)

Streaming / Processing on Interactive Timescales

- Experiments are of relatively short duration (1-5 days)
- Frequent changes to complex experimental setups and device configurations
- Need to debug / optimize setups without resorting to detailed offline analysis
- **Require a streaming readout and significant inline processing with seconds of latency**



Projection of gamma-ray interaction points from a collimated source derived from trace data.

GRETA Experimental Requirements

- Computing scale set by the **online** signal processing requirement (signal decomposition)
  - 480 k events / s **(objective KPP - CD4A requirement)**
  - Implies 4 GB / s aggregate post FPGA processing - 8 kB / event (when compressed)
  - Base calculation requires ~5 ms / signal decomposition - requires large, detector-specific, in-memory simulation (~ 2 GB each)
  - Each experiment can load the cluster differently** - up to 7:1 rate asymmetry in detector rates
  - Latency from electronics → online analysis is ≤10s **(Time-sensitive requirement)**

- Implies:
  - A mid-size cluster (several thousand cores needed to just keep up)
  - A flexible way to allocate cluster resources to detector needs**

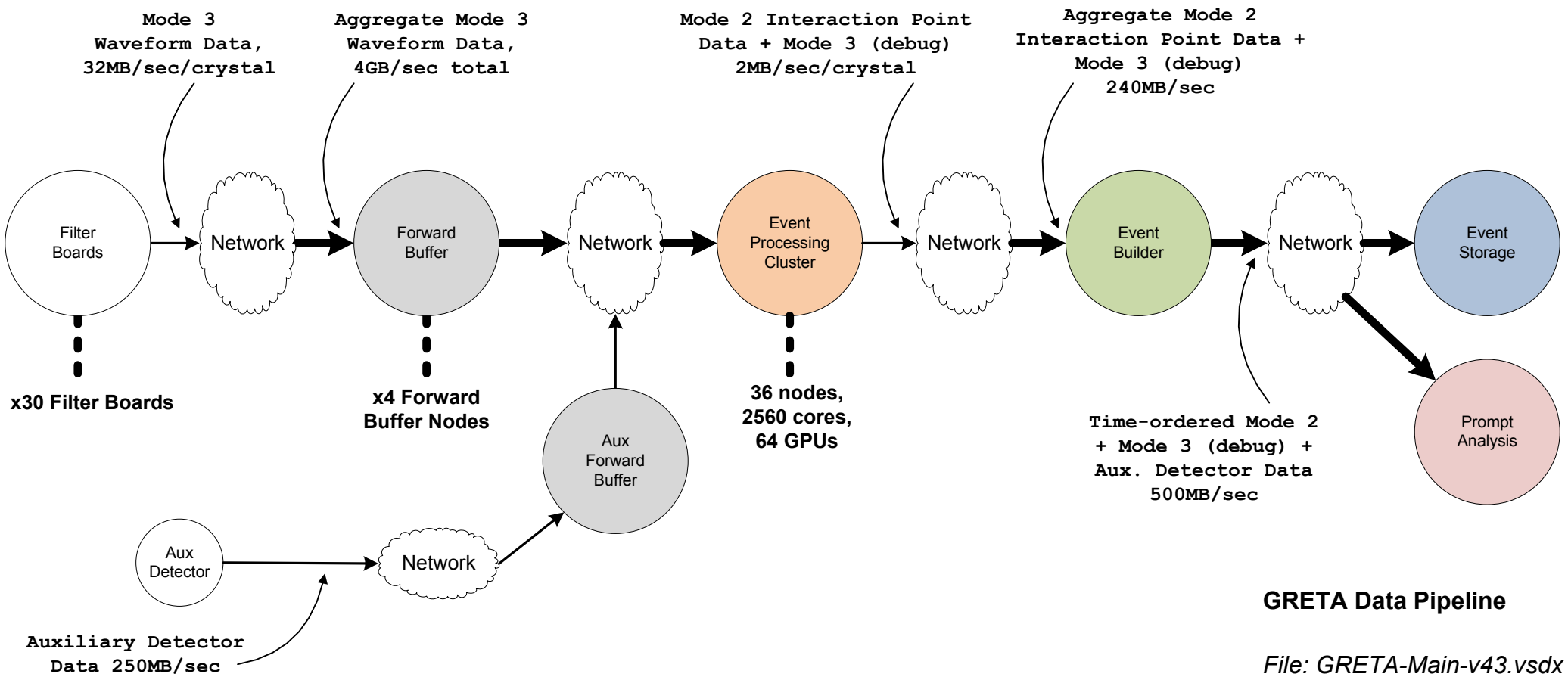
- Key issue:
  - How to connect ~30 large FPGA's (Xilinx Ultrascale+) with 4800 FADC-fed signal processing pipelines to computing resources to perform this calculation? (with a handful of people working part-time, limited budget)

Network-based Architecture

- Both computing **and electronics** components are interconnected by a high-speed network *(no explicit 'bus-based' interfaces)*
- Data from electronics boards as a stream of UDP packets
- All computing components are networked Linux-based systems, software deployed as containers with HTTP-based APIs used throughout

- Advantages:
  - Standard components (Ethernet switches, commercial server and cluster hardware, etc.)
  - Standard software (nanomsg, docker, prometheus)
    - No need to re-invent the wheel
    - Significant productivity gains
    - Ease future maintenance of the system

Pipeline-Based Network Architecture



GRETA Data Pipeline

File: GRETA-Main-v43.vsdx

See 'Simple and Scalable Streaming: The GRETA Data Pipeline, EPJ Web of Conferences 251, 04018 (2021)'

Key Pipeline Components

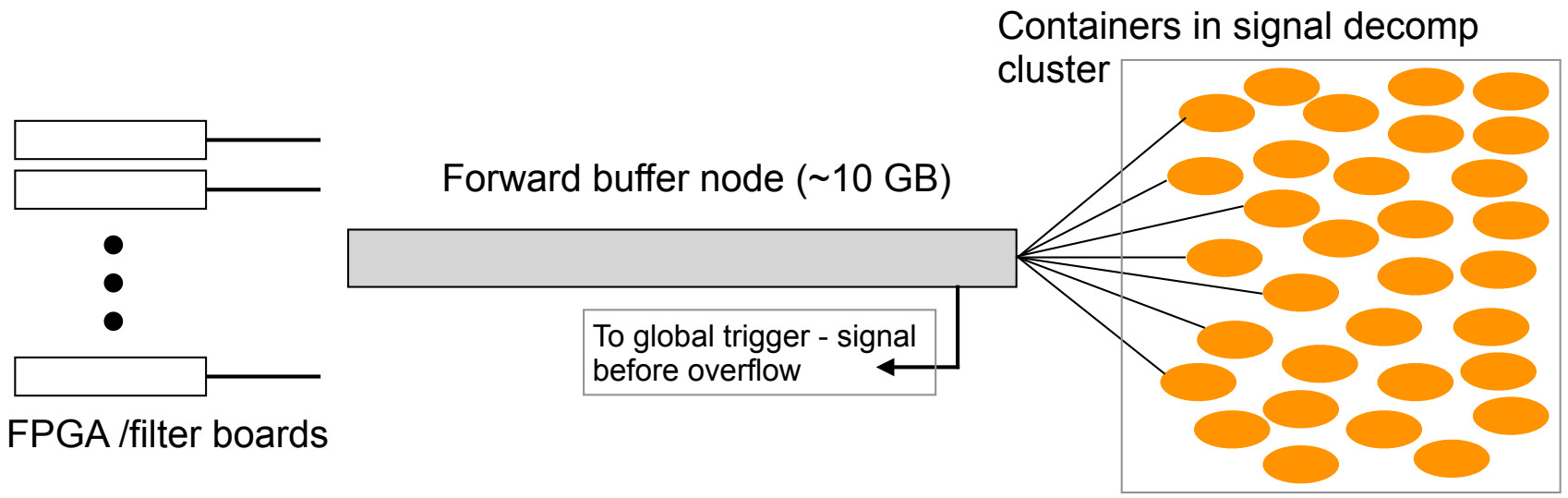
- **Forward Buffer:**
  - Mediates communication between the electronics and computing cluster
  - Allows electronics to be free running and cluster nodes to self-schedule
- **Event Processing:**
  - Locate interaction points from waveforms (signal decomposition)
  - Each compute node presents a number of job slots which are assigned to a specific detector
  - Hybrid CPU/GPU implementation
- **Event Building:**
  - Aggregates events according to timestamps into global events
  - Provides a tap for online data monitoring

- Pipeline components transfer data via GAP (Greta Application Protocol) layer, based on nanomsg - zero copy, low latency, excellent stability (https://nanomsg.org)

Transport and Physics aspects are well separated in the protocol - simplifies application to other experiments / application domains

Forward Buffers

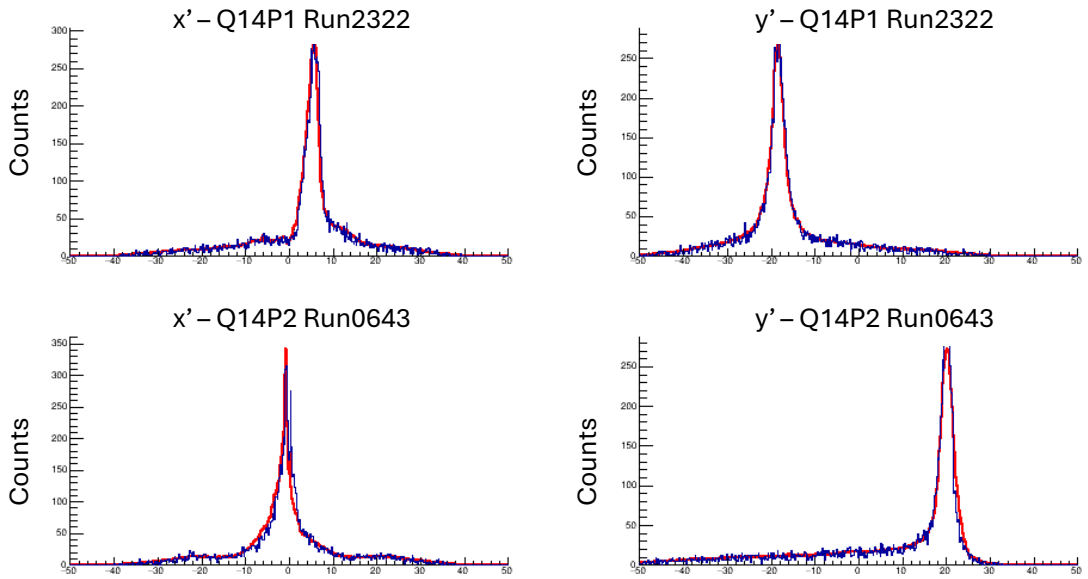
- Forward buffer is central to GRETA's computing architecture
- Aggregates data from electronics, serves data for event processing:
  - Allows electronics to be free running - have a simple 'push' interface using UDP
  - Event processing containers can implement a 'pull' interface and self-schedule
  - Fill state of forward buffer queues allows implementation of global flow control



This architecture greatly simplifies the FPGA readout implementation; allows use of standard networking/computing hardware and software tools

Performance - KPP Test Results

- Test ran for an hour over which time 1.85 x 10^9 compressed waveform messages were processed
- Single file of 636 GB of interaction point messages was generated
- No events lost in signal decomposition or transport through the pipeline:
  - 1857633780 events at forward buffer ingress, 1857633780 decomposed events in file
- Aggregate rate of 510725 signal decompositions / s - meets objective KPP for computing

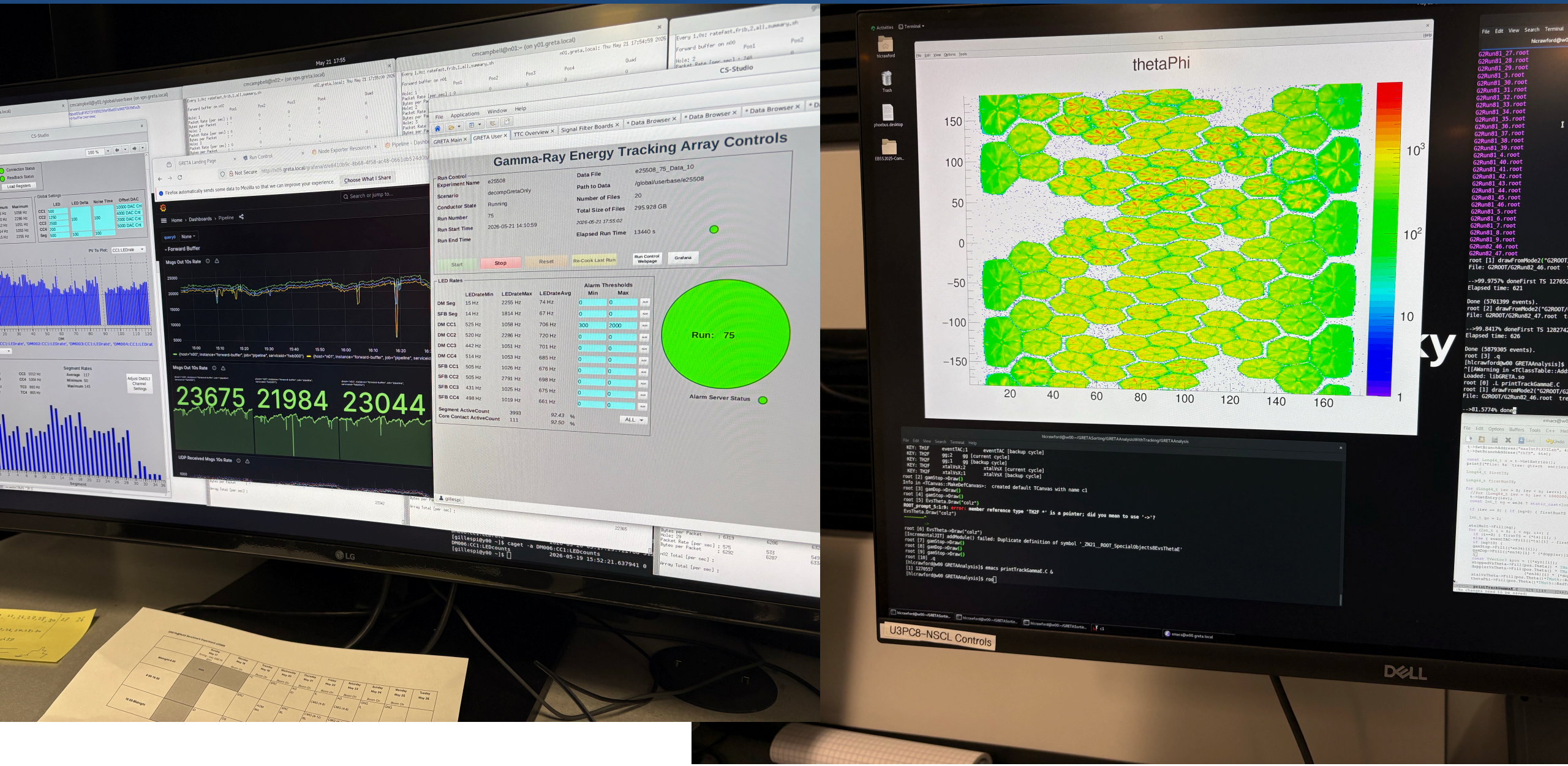


x-y projections: blue (GRETA - 10k events), red (GRETINA reference set)

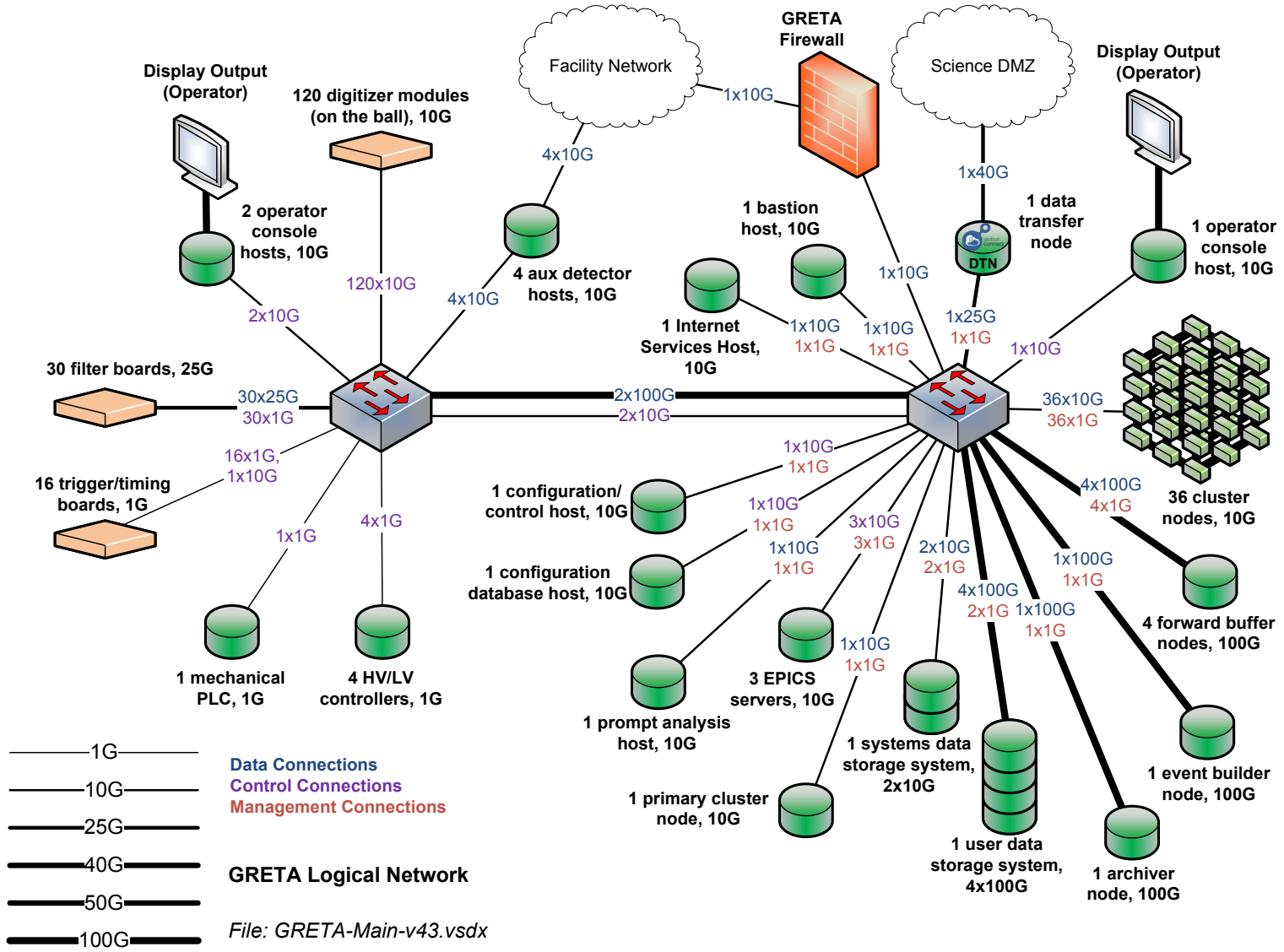
Forward Buffer	Start (GMT)	Stop Request (GMT)	Stop (GMT)	Duration (s)	Total Msg In	Total Msg Out	Rate (Msg/s)
n00	23:09:53	0:10:30	0:10:30	3637	463775188	463775188	127516
n01	23:09:53	0:10:30	0:10:30	3637	465339834	465339834	127946
n02	23:09:53	0:10:30	0:10:31	3638	463581217	463581217	127427
n03	23:09:53	0:10:30	0:10:30	3637	464937541	464937541	127835
Aggregate					1857633780	1857633780	510725

Statistics gathered from a one hour test utilizing event traffic simulators and 4 forward buffers in KPP test

GRETA Commissioning Run, 5/26



GRETA's Local Computing Infrastructure





Can we offload some of this complexity?

Geographically Distributed Pipelines

- Changes in landscape:
  - Advances in optical networks -100Gb local switched networks common-place, >400Gb wide area networks (ESnet)
  - HPC facilities provide unprecedented compute capability

- The Idea - Distribute pipeline components on multiple experimental and HPC facilities
  - Avoid the need for complex hardware and software deployments at experiments - make it cheaper, faster
  - Send events (not files) with seconds of total latency
  - Make pipelines ‘composable’ across facilities so pipeline components can run where resources are appropriate (compute, storage)
  - Enables the application truly large computing resources to real time steering of experiments

GRETA/Deleria Project

- **Goal:**
  - *Demonstrate high-bandwidth, event-by-event readout with significant inline processing at remote HPC facilities (OLCF/ORNL) on interactive timescales.*

- Basic idea is to `externalize` GRETA's forward buffers - serve as edge devices to national wide area networks

- Multi-area LDRD (Physical Sciences and Computing Sciences at LBNL) in collaboration with staff at OLCF/ORNL

Deleria - Distributed event-level experimental readout & inline analysis

Testbeds we employ

- ACE IRI testbed (ORNL)
  - Use the Defiant cluster (Frontier demonstrator)
  - Use Slurm (via JANIS/DTNaaS) for container start, Greta/Deleria conductor for pipeline config

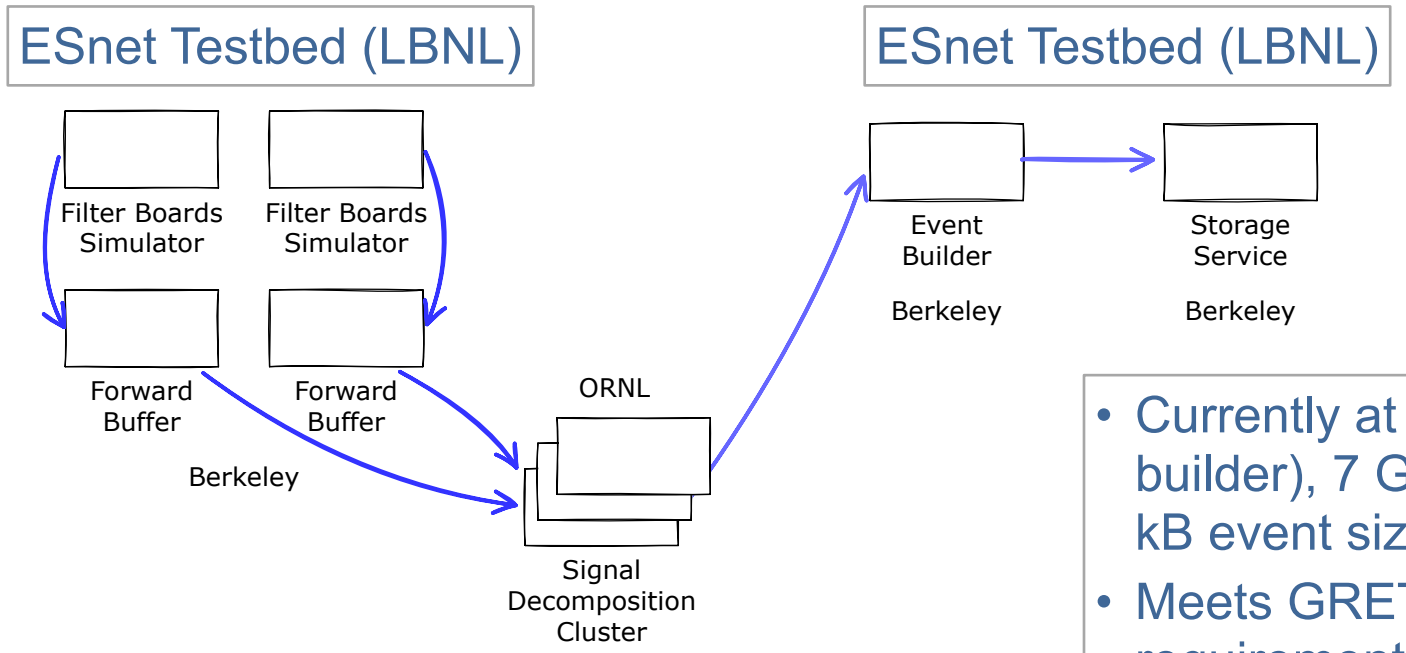
- IRI ESnet testbed (LBNL)
  - Nodes to run non-computational intensive pipeline components + high-performance storage
  - Uses JANIS/DTNaaS for container management

- Established layer 2 connection between the two systems (currently 40 Gb/s, can be upgraded to 100 Gb/s)

- Geographical separation allows us to test the impact of high-latency links (RTT = 110ms)

Testing/Optimization

- Utilize GRETA filter board simulators to replay experiment events into forward buffers → OLCF for processing
- Use Deleria-decomp passthrough so we could model calculation latency.



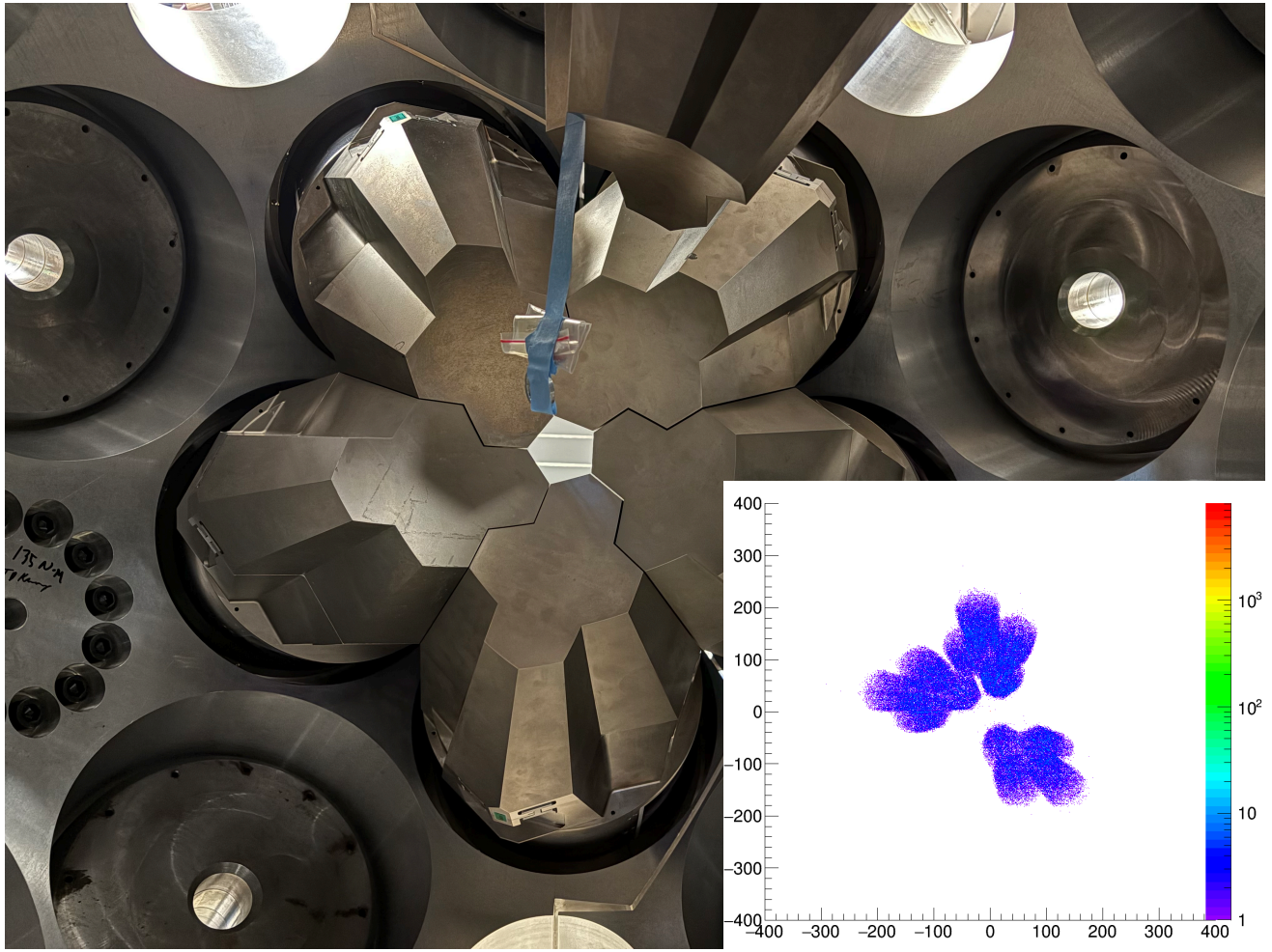
RTT = 110 ms

~50k events in-flight at GRETA design rate 🙂

- Currently at 35 Gb/s (to event builder), 7 Gb/s (to storage), 8 kB event sizes
- Meets GRETA's design requirement for bandwidth (32 Gb/s)

Testing with Detectors

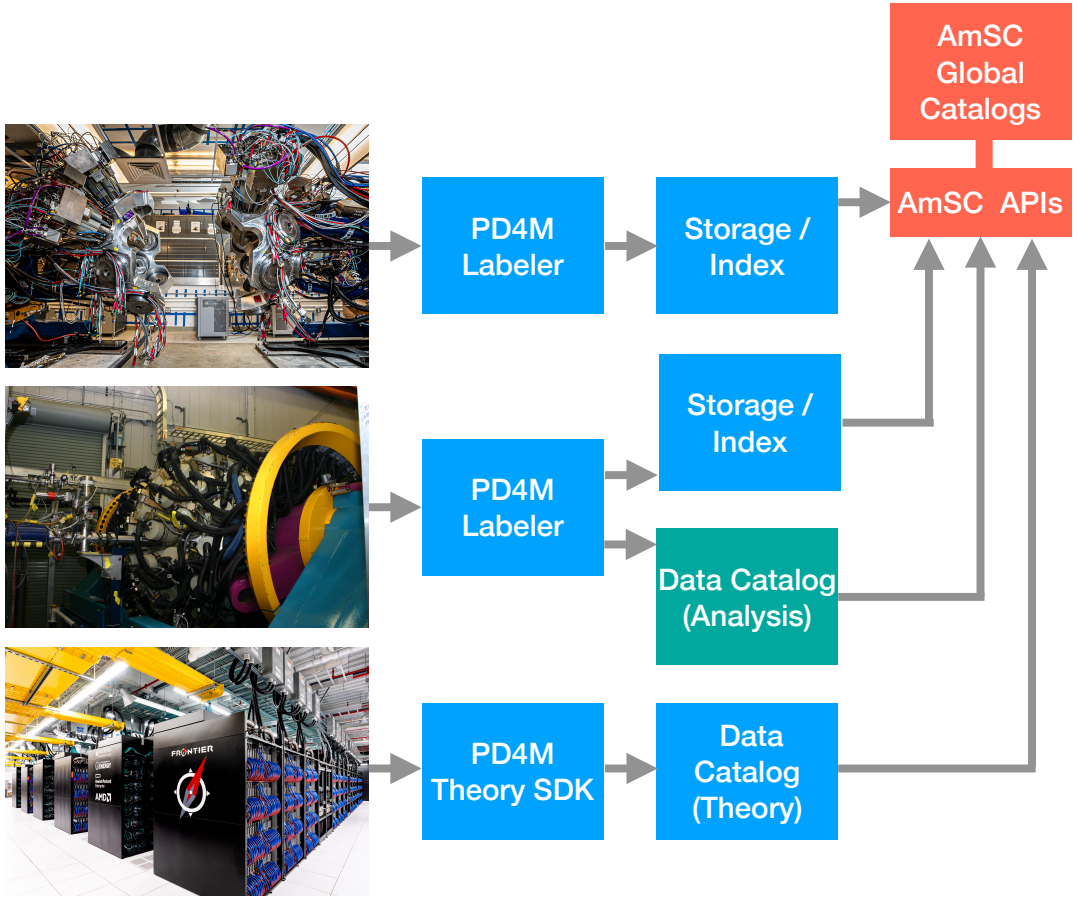
- During GRETA system assembly at LBNL, we streamed source data from the 3 instrumented quads available from the 88 K-Area through the ESnet testbed, to ORNL for interaction position reconstruction and back
- Plan a second test processing source data collected at FRIB and streamed to Defiant at ORNL with full complement of GRETA quads (30)



Data streamed from 3 quad detectors during GRETA system assembly at LBNL to ORNL - subpanel shows distribution of interaction points

AI Applications - PD4M

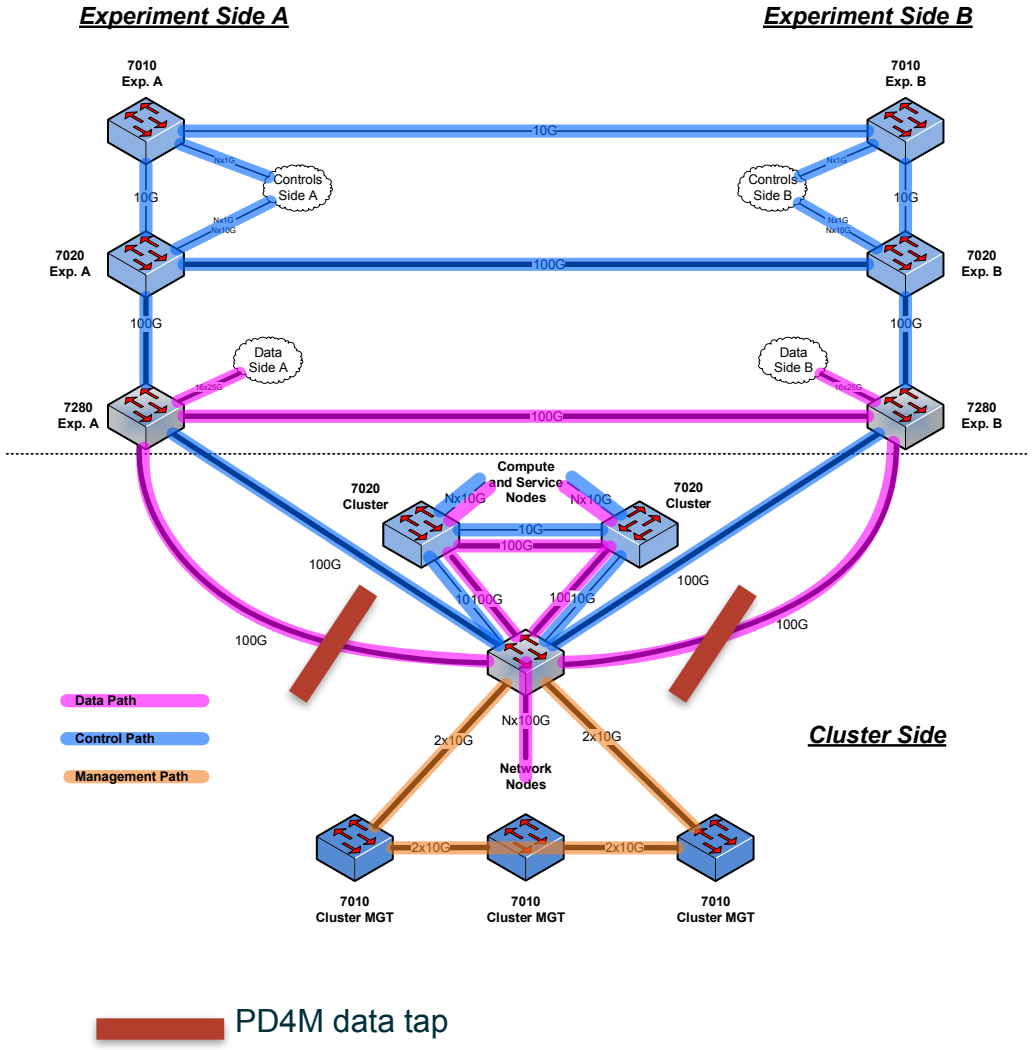
- Labelers are a software component running on a machine local to the experiment or theory calculation
- Act as a data ‘tap’
- Associates labels with raw data and associated context (controls, experimenter annotations, theory workflows)
- Labelled data / metadata formatted and stored locally or streamed to AmSC resources
- Project part of the DOE Genesis mission (DaPP)



Block diagram of PD4M showing experimental devices (GRETA, Gammasphere) and theory codes (Frontier) with PD4M labelers, the theory SDK, data catalogs with the AMSC API.

Relationship to Data Streaming

- Idea is to have an “always on capability” - all data collected is used for future training or online inference to control and analyze the experiment
- Labelers tap data stream at their source (at or close to electronics subsystem)
- Labeler sees a duplicate of the experiment’s data stream (GRETA - switch port mirroring, Gammasphere - software mirroring)
- Designed to have minimal impact on experiment operations
- For real time inference and feedback into experiment performant data streaming is necessary



GRETA network diagram showing PD4M taps on core data path

Some Lessons Learned

- Important to build to a physics-based specification
  - Constrains architecture choices
  - Forces one to consider the full scope of the experimental problem
- Multidisciplinary team is essential
- Orchestration problem was underestimated

Summary

- Developed and deployed a streaming readout system for GRETA capable of running at high rates with significant amounts on inline compute operating on interactive timescales
- Software based forward buffers allow electronics to be free running while cluster-based processing to self-schedule
- Demonstrated streaming pipeline can work over high-latency WAN links at both high message rates and bandwidth enabling online remote processing at HPC facilities
- Applying techniques to training and real time inference pipelines in support of the Genesis mission

The Team

- Eric Pouyoul, Eli Dart, Kiran Vasu, Ezra Kissel, Seyoung Yu - ESnet, LBNL
- Gustav Jansen - NCCS / Physics, ORNL
- Tin Ho, John White - Science IT group, LBNL
- Tynan Ford - Engineering division, LBNL
- Ross Miller - OLCF, ORNL
- Mario Cromaz - Nuclear Science Division, LBNL