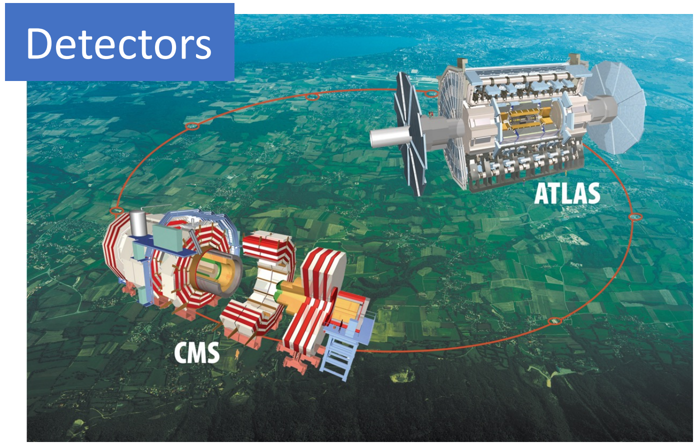
Service for Optimized Network Inference on Coprocessors (SONIC) in CMS and ATLAS

Yao Yao

March 1st , 2024

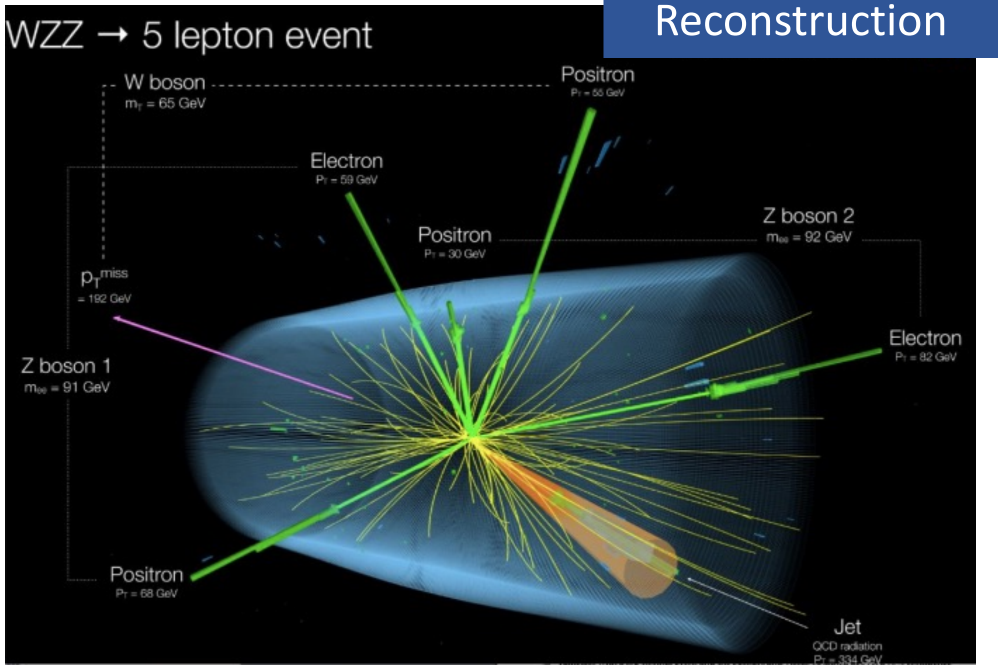
Introduction to CMS and ATLAS

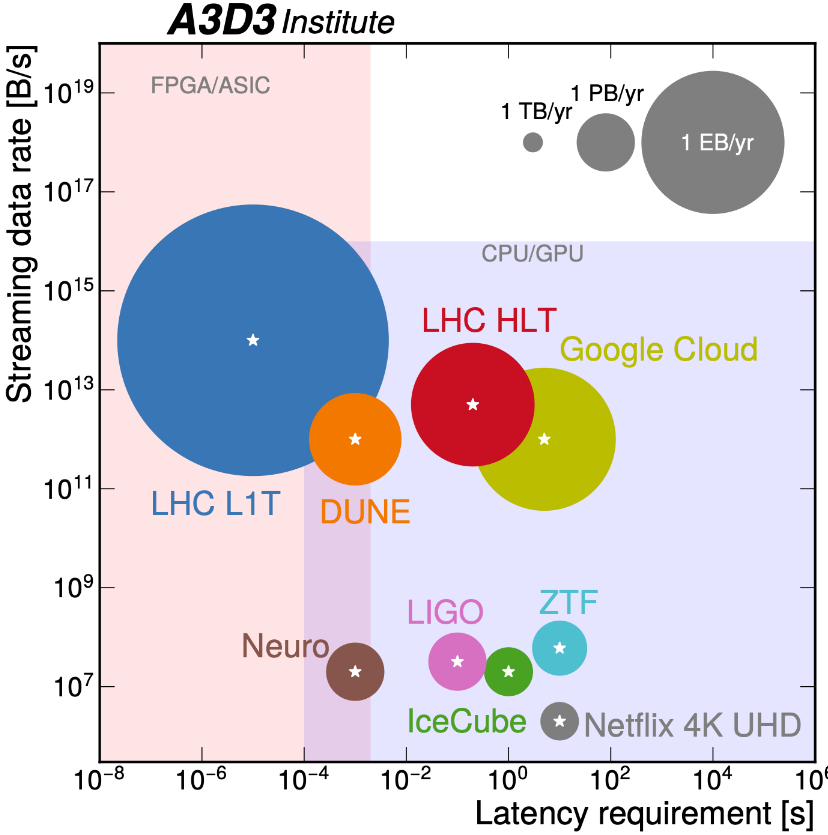
Detectors



- Proton-proton collisions happen every 25 ns.
- Immediate decision of which events to store (online L1T + HLT).
- Full reconstruction of each stored event (offline).

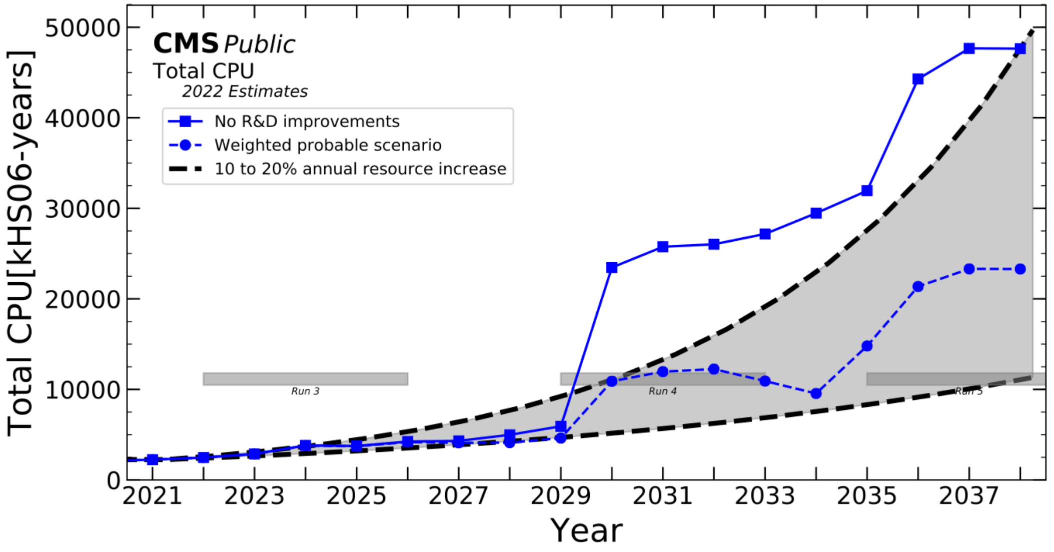
Reconstruction

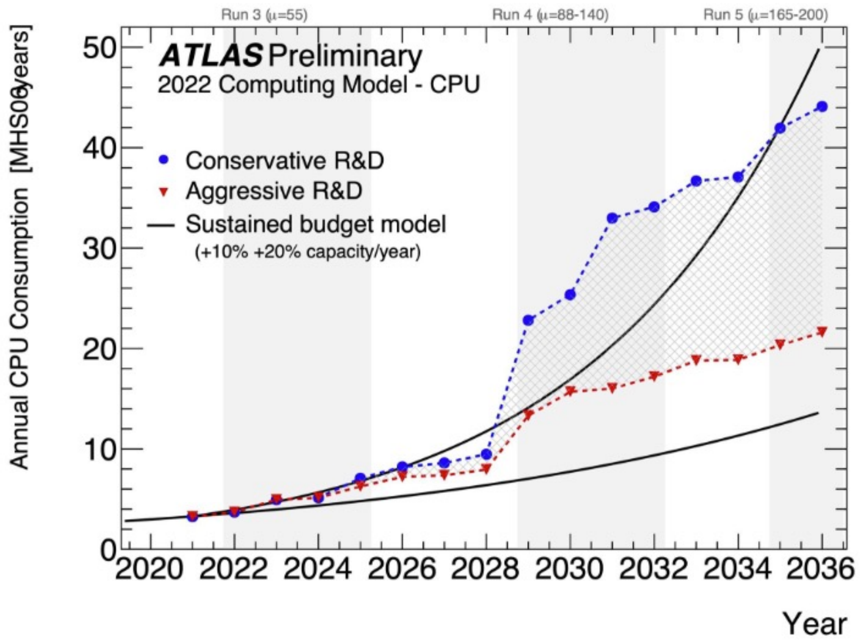




Challenge on data processing

- To process the data being collected in the HL-LHC, the data processing workflow for both ATLAS and CMS experiments are expected to be more complicated.

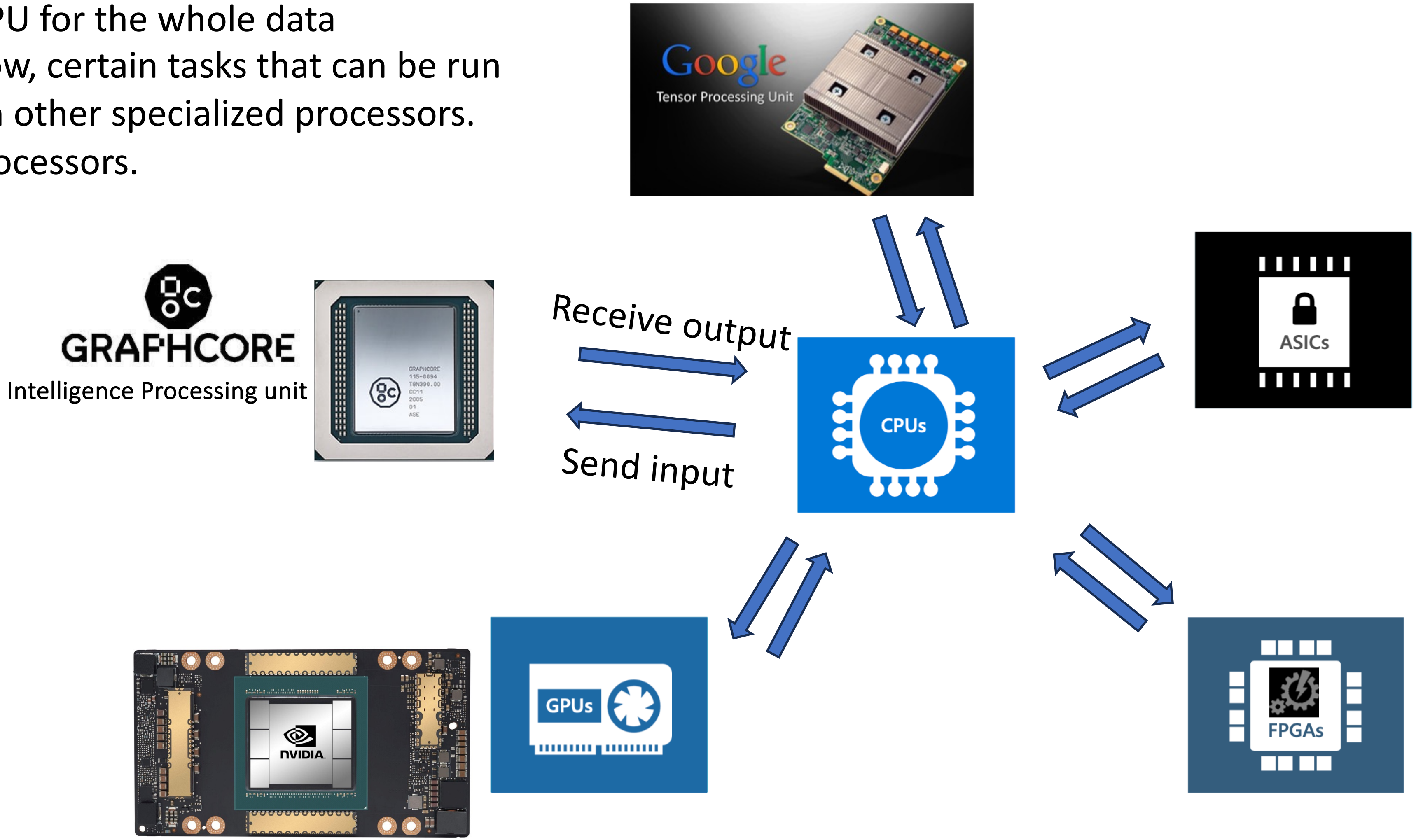




- To enhance data processing ability with limited computing resources, we need to explore a way that fully utilizes the computing resources that are accessible and provide a fast and reliable data processing workflow by Run 4.

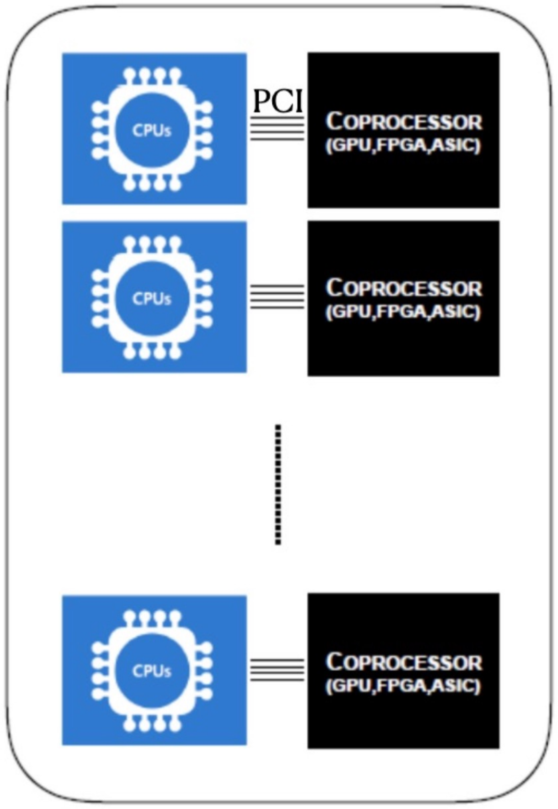
Heterogeneous computing platform

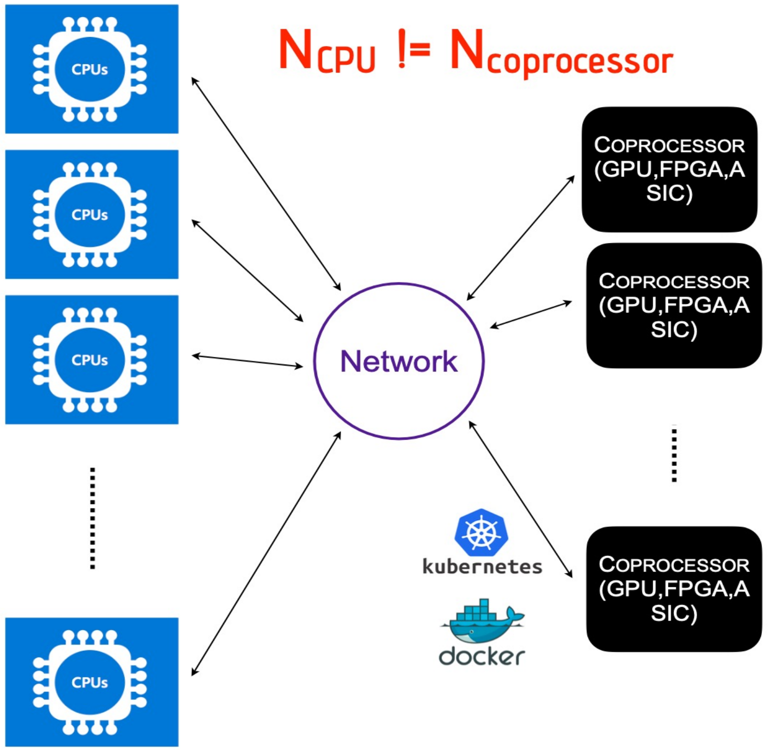
- Instead of using CPU for the whole data processing workflow, certain tasks that can be run more efficiently on other specialized processors. We call them coprocessors.



Inference as-a-service (Iaas)

- There are two ways to realize this cross-platform data processing:
  - Direct connection between CPUs and coprocessors. Advantage: fast and stable. Disadvantage: not flexible and not fully utilized due to inferences' complexity varies.
  - Inference as-a-service. Advantage: flexible and CPU-coprocessor ratio can be optimized. Disadvantage: network topology and stability affect the inference throughput and latency.





SONIC with different coprocessors

Studies that demonstrate the feasibility of physics computing with SONIC through FPGAs, GPUs, IPUs, and TPUs(on-going)

FPGA: on *Microsoft Brainwaves.* arXiv:1904.08986v2

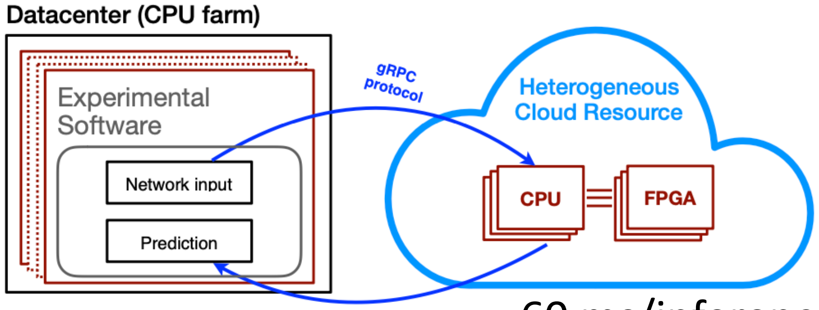
- The study used ResNet-50 and trained a model to perform top quark tagging.

GPU: on *Google Cloud* arXiv:2007.10359v2

- The study demonstrated a framework that enables the Deep learning inferences to process LHC data on GPU servers.

GPU and GraphCore IPU on *Google Cloud, Purdue Tier 2, and IPU team* arXiv:2402.15366

- The study realizes multiple algorithms being run on the GPU server in the CMS MiniAOD data production workflow



60 ms/inference
client: Fermilab
server: Virginia

Inference as-a-service (Iaas)

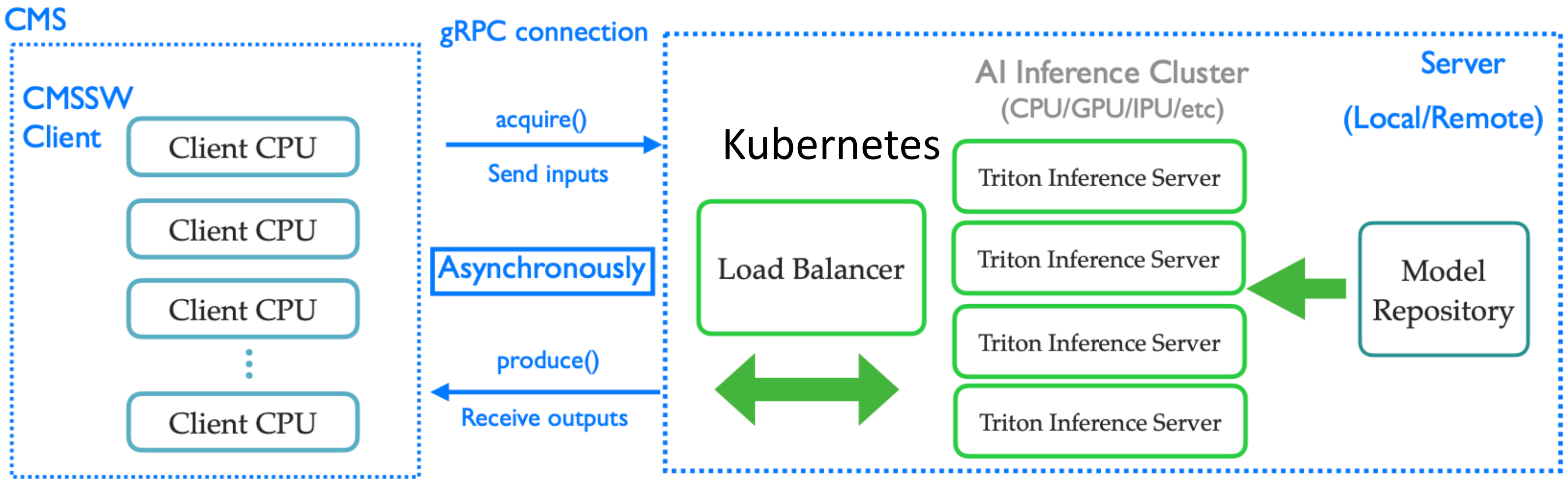
- We encounter two scenarios when adapting the current data processing workflow to heterogeneous computing platform:

Scenario 1:
physics algorithms that can be re-casted as machine learning problem
Approach:
use the supported backend for new hardware

Scenario 2:
physics algorithms that are CPU based and not ML, but still can be accelerated on certain co-processors
Approach:
re-write physics algorithms for new hardware

SONIC in CMS

MiniAOD workflow with SONIC is realized on GPUs through Nvidia Triton server.
Triton supports many ML backends: ONNX, TensorFlow, PyTorch, Scikit-Learn.



Data tier	Event size [kB/event]
Raw	1000
AOD	480
Mini-AOD	35–60
Nano-AOD	1–2

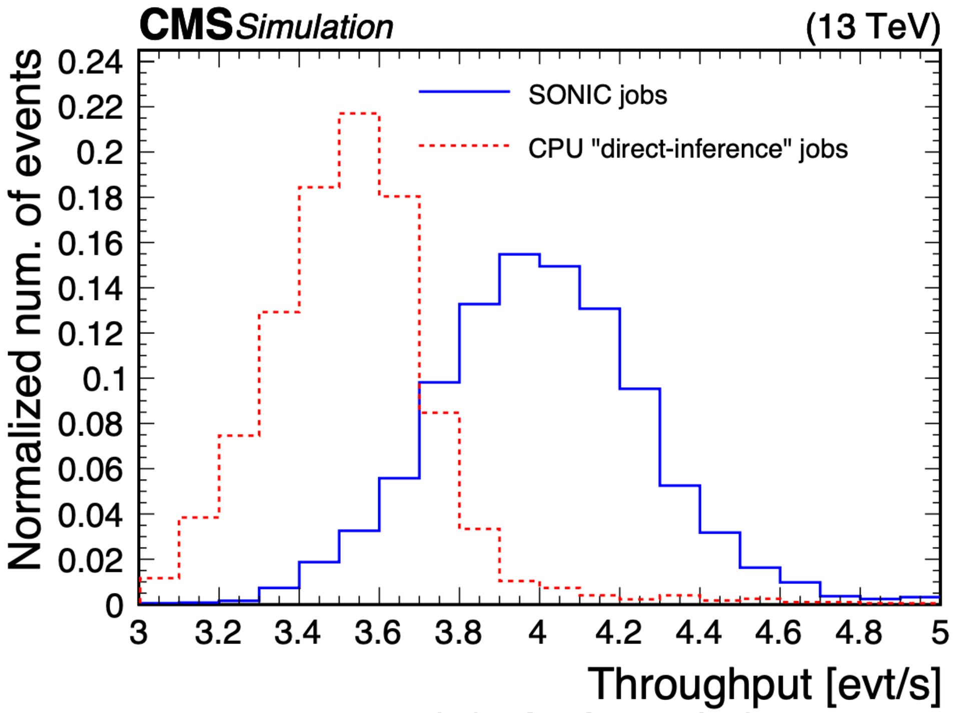
Algorithm
PN-AK4
PN-AK8
DeepMET
DeepTau
ParticleNet+DeepMET+DeepTau
Full workflow

Developed:
MiniAOD Run II workflow
Developing:
MiniAOD Run III workflow with the more ML algorithms.

Performance and Benchmarking

MiniAOD RunII workflow:

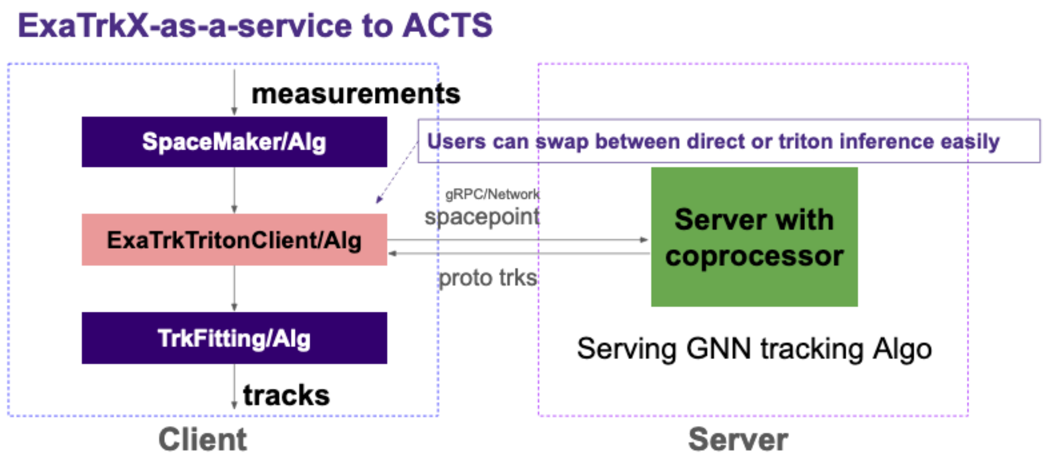
- Per-model optimization is accomplished with Triton model analyzer tool.
- Cross-site tests to measure the latency of the network.
- Large Scale test for the whole workflow with big mount of simultaneous client-side jobs. (see figure)
- CPU fallback servers: make sure the performance is not worse than CPU directly inference

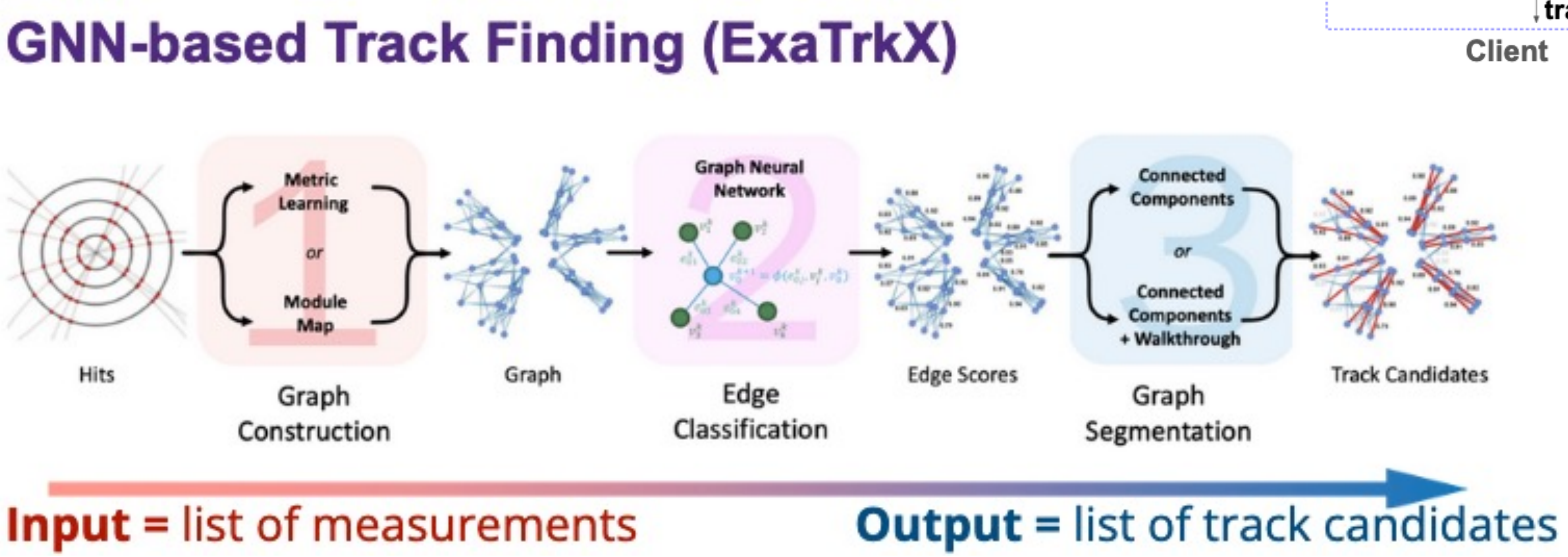


SONIC in Tracking with ACTS

ACTS is an experiment-independent toolkit for (charged) particle track reconstruction in (high energy) physics experiments implemented in modern C++ and can be adapted to any tracking detector.

ATLAS is planning to use ACTS to replace the current tracking modules.





Non-ML SONIC in CMS

Charged particle track reconstruction is the most expensive and the most time-consuming step in the object reconstruction pipeline.

Developed Patatrack-AAS:

- Patatrack is accelerated pixel track reconstruction in the CMS.

Developing Line Segment Tracking AAS for HL-LHC in the CMS.

Both algorithms are highly parallelable and are already written in the way that *can be run on GPUs*.

To be developed:

1. Automated ALPAKA backend
2. SONIC in High Level Trigger (HLT) computing farm.

Non-ML SONIC with ACTS

trac-cc is a project that rewrites the ACTS algorithms such that they can be run on GPUs

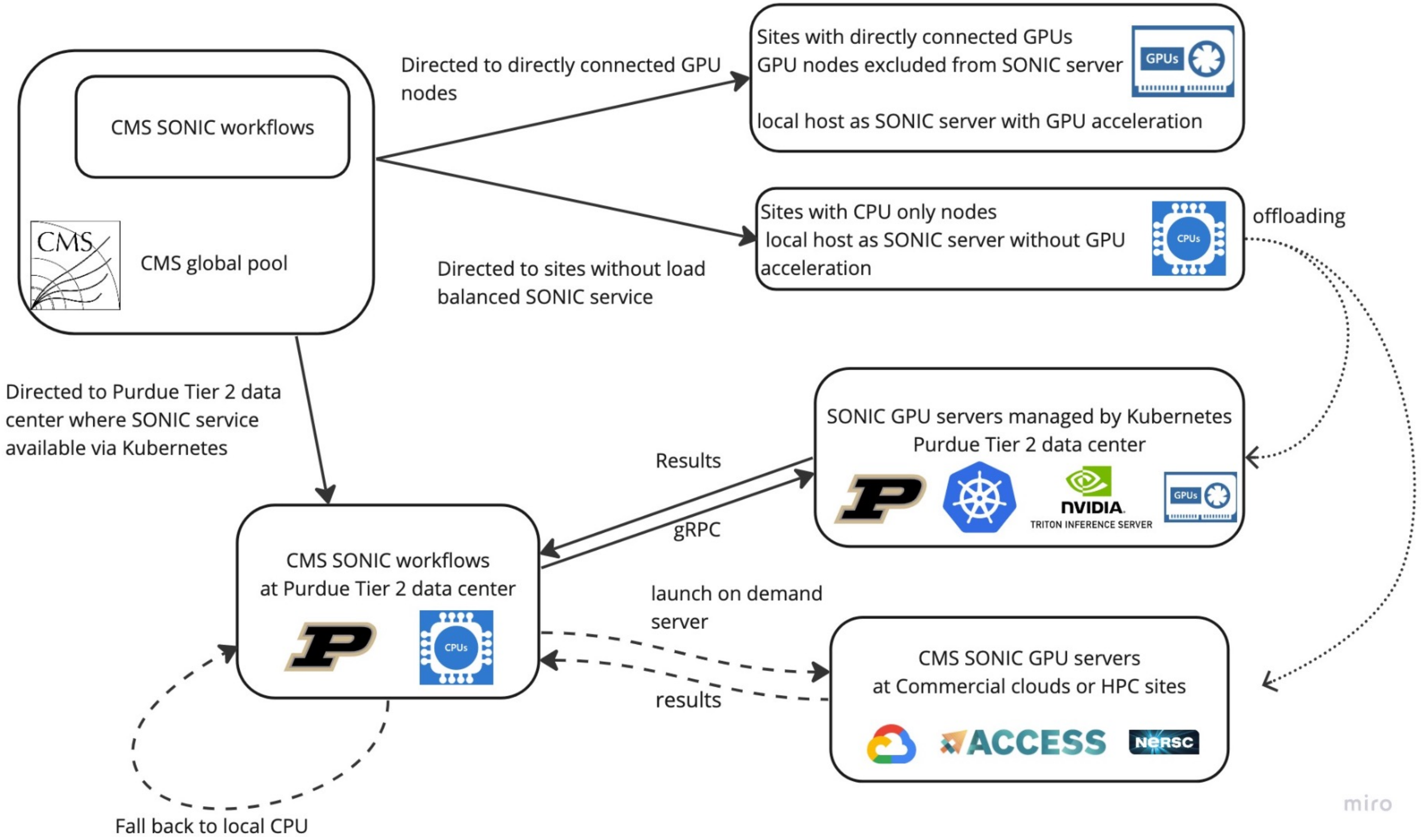
trac-cc as-a-service *in the future*?

Features

Category	Algorithms	CPU	CUDA	SYCL	Futhark
Clusterization	CCL	✅	✅	✅	✅
	Measurement creation	✅	✅	✅	✅
Seeding	Spacepoint formation	✅	✅	✅	⚪
	Spacepoint binning	✅	✅	✅	⚪
	Seed finding	✅	✅	✅	⚪
	Track param estimation	✅	✅	✅	⚪
Track finding	Combinatorial KF	✅	✅	🟡	⚪
Track fitting	KF	✅	✅	✅	⚪

✅: exists, 🟡: work started, ⚪: work not started yet

SONIC in CMS central production



Summary

Both CMS and ATLAS (ACTS) has been working on developing SONIC for both ML and non-ML algorithms.

Developed/Developing	CMS	ATLAS (ACTS)
ML algorithms	Tagging algorithms in MiniAOD	Exatrkx
non-ML algorithms	Patatrack Line-segment tracking	More algorithms on ACTS

There are many details in the service implementation and examination, including

- Backend development for different algorithms for the coprocessor hardware.
- Per model optimization, batch size, batch window, etc.
- To saturate the coprocessors, multiple models being launched on servers, evaluate CPU to GPU ratio, load balancing with Kubernetes.
- Latency and throughput.

We consider the real scenarios in productions since we try to benefit the CMS and ATLAS workflow with SONIC, there are more things to consider: server set up in multiple sites, big scope load balancing, latency, fallback options.