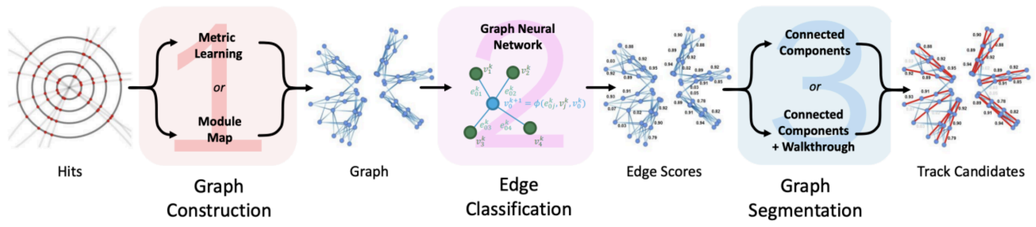
Graph Neural Network-based particle tracking as a Service

- **Graph Neural Network (GNN)**-based algorithms (**ExaTrkX**) could be effective for finding track candidates in ITk
  - Its computational requirements present significant challenges
  - Very slow inference on CPUs → GPU-base acceleration will be crucial
- Every site will not have GPUs → We propose to run this algorithm using **as a service computing model**
- **Current tests show that we can achieve higher throughput by running ExaTrkX as a service**

Elham E Khoda[1], Andrew Naylor[2,3], Xiangyang Ju[2], Steven Farrell[2,3], Shih-Chieh Hsu[1]

[1]University of Washington, [2]LBNL, [3]NERSC,

GNN-based TracK Finding: ExaTrkX



Input = list of *space-points* **Output =** list of track candidates

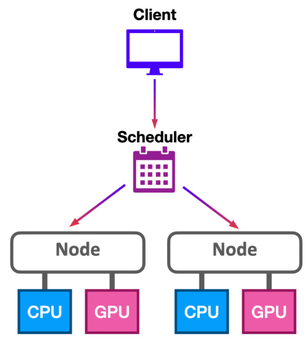
A tracking graph: nodes are space-points and edges are possible connections between nodes.
- True edges are connections of nodes from the same particle of interest

- Similar efficiency as the classical algorithm, and $\mathcal{O}(10^{-3})$ fake rates (200 pileup)
- Can be accelerated on different coprocessors (eg GPUs)

As a Service Computing Model

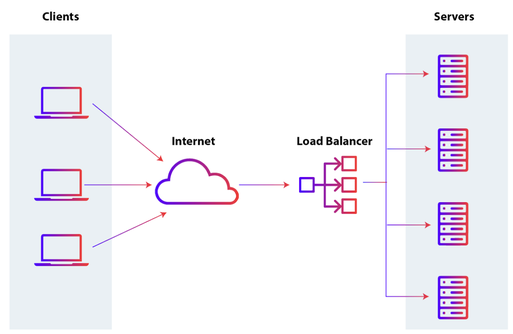
Direct Connection

CPUs and GPUs are connected



As a Service

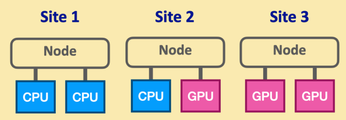
No need to have a local GPU



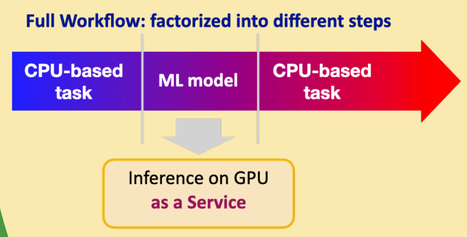
- **Client - Server** connections are made through network
- Server running on single / multiple GPUs
- Single server can process multiple client requests

Why as a Service ?

Every Computing site will not have GPUs



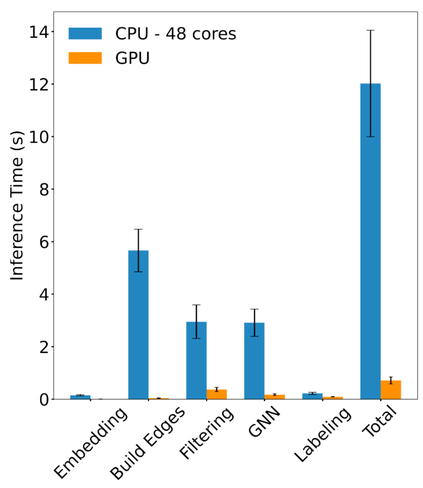
- Adding GPUs to existing CPU only sites is expensive



- Portable solution for supporting different coprocessors
- Factorizing out ML workflows
- Potential for more efficient use of coprocessors thanks to the scheduling capabilities in the Server!
- CMS have already seen positive results with SONIC using **Nvidia's Triton Inference Server**

ExaTrkX timing

GPU (V100) inference is ~20x faster than 48 CPU cores

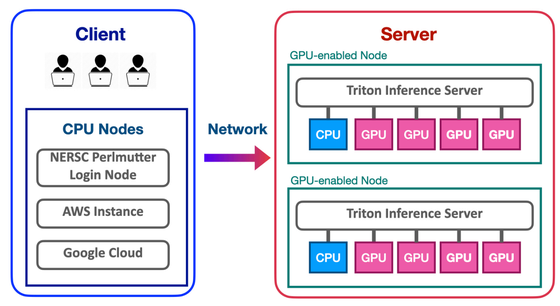


⇒ GPU is crucial to run GNN-based track finding algorithm

Figure: Inference Time on GPU and CPU (48 cores)

ExaTrkX as a Service

Stand-alone ExaTrkX tool



GPU stress tests on NERSC Perlmutter, AWS and Google Cloud virtual machines

- Client and Server running on separate nodes
- **Server:** Nvidia Triton Inference Server

NVIDIA TRITON INFERENCE SERVER

Table: Latency tests on Perlmutter (up to 4 A100 GPUs in a node)

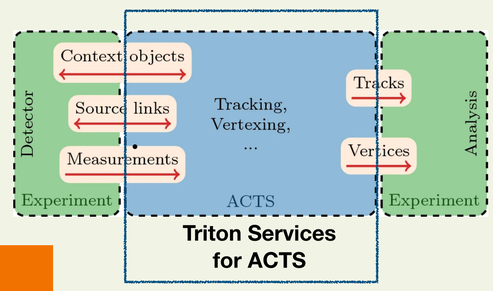
Mode	Max number of requests	Total Time [s]	Time / event [ms]	No. of GPUs in use
Direct	10	19.79	19.79	1
AAS: Pytorch	40	104.94	26.24	4
AAS: All Models	128+	312.27	24.4	4

Direct Connection Vs As a Service (AAS) approach

⇒ Similar per event latency

AAS approach can handle more client requests

Future: ACTS as a Service



Summary

- **Successful setup of ExaTrkX as a service on Perlmutter HPC and clouds**
- AAS approach can handle more client requests
- **We are integrating the ExaTrkX as a service into two major tracking frameworks: ACTS and Athena**
- Preliminary implementations are done → to be validated, tested and optimized

References

- Graph Neural Networks for particle reconstruction, NeurIPS ML4PS (2019)
- ExaTrkX performance or HL-LHC particle tracking: EPJC 81, 876 (2021)
- GPU Coprocessors as a service, Mach. Learn.: Sci. Technol. 2 (2021) 035005
- Accelerating ExaTrkX inference, J. Phys.: Conf. Ser 2438 012008 (ACAT 2021)
- CMS Mini-AOD Production with Coprocessors as a Service (CHEP 2023)
- A Common Tracking Software, Comput.Softw.Big Sci. 6 (2022)