# Scaling ML Introduction

Paolo Calafiura & Walter Hopkins

# Introduction

- DOE computing facilities will remain GPU focused
  - Attractive for AI/ML workloads
  - Several ML models developed by the HEP community significant resources (e.g., take days to train)
- The Scalable AI/ML (SMiLe?) area will
  - Select 2-3 large-scale ML models representative of HEP
  - Using these testbed models, we will then select and optimize suitable scalable training solutions on HPC systems.
  - The work will require the evaluation of distributed data-parallel ML training components, as well as hyperparameter optimization tools .
- The optimization of the associated parallel I/O will benefit from past HEP-CCE experiences.

# Scope

- Performance studies of ML pipelines (both for training and inference) on heterogeneous resources
- Shorten the development cycle for two large models weeks→hours
- Develop a CCE distributed ML platform adopted by at least two experiments

Scope does not include developing or fine tuning ML models themselves: models to be tested should be “production” ready

# Inference as a Service (IaaS)

Why is it useful?

- Clean interface to abstract host/device communication, even across WAN (study performance impact, usability, security on DOE HPC systems)

What we would do within SMiLe?

- Similar plans as for to training, but not necessarily the same models?

Opportunity to collaborate with PAW on workflow optimization

# Proposed Milestones

1. Identify target ML models in collaboration with experiments
2. Port, train, and run at least two target models on two different HPC systems
3. Compare two data parallel training solutions for at least one target model
4. Compare two hyperparameter optimization tools on at least one target model
5. Setting up a prototype Inference as-a-service platform on at least one DOE HPC system

# Possible target ML models

- Particle ID
  - Flavor tagging in ATLAS: training takes days, HPO not tried due to computing limitations
  - Particle flow (full particle classification with ML). Based on transformers and very compute intensive training
- Tracking
- Likelihood free inference

Any ideas (especially from Cosmic and Intensity frontier) are very welcome!