Scalable AI/ML Workflow Management Across Distributed Heterogeneous Resources With PanDA

Wen Guan, Tadashi Maeno, Rui Zhang (WISC), Xin Zhao, Christian Weber and Torre Wenaus on behalf of the PanDA team

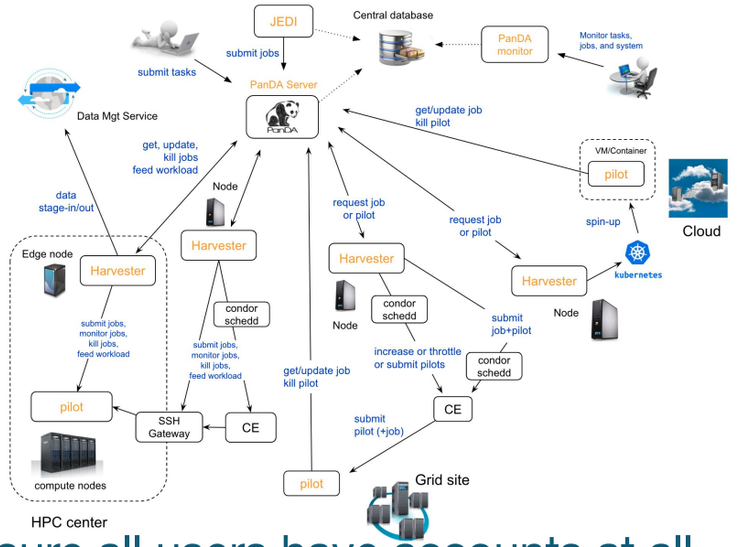
AI4EIC 2023 Annual Workshop

Nov 30, 2023

Introduction

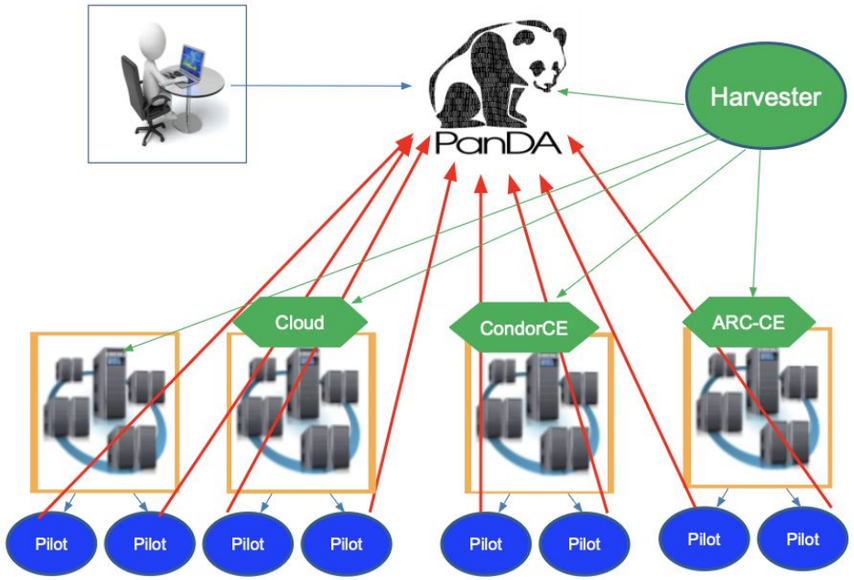
- Scalable and Distributed Computing with PanDA/iDDS
  - PanDA: Workload management across distributed heterogeneous resources
  - iDDS: complex workflow orchestration
- Distributed ML with PanDA/iDDS
  - Use cases:
    - HyperParameterOptimization (HPO)
    - Monte Carlo Toy based Confidence Limits
    - ActiveLearning
- Adapt PanDA/iDDS for AI-assisted Detector Design at EIC
  - Distributed Optimization with AID(2)EIC
  - Adapt to AID(2)EIC with PanDA/iDDS

PanDA (Production and Distributed Analysis system)



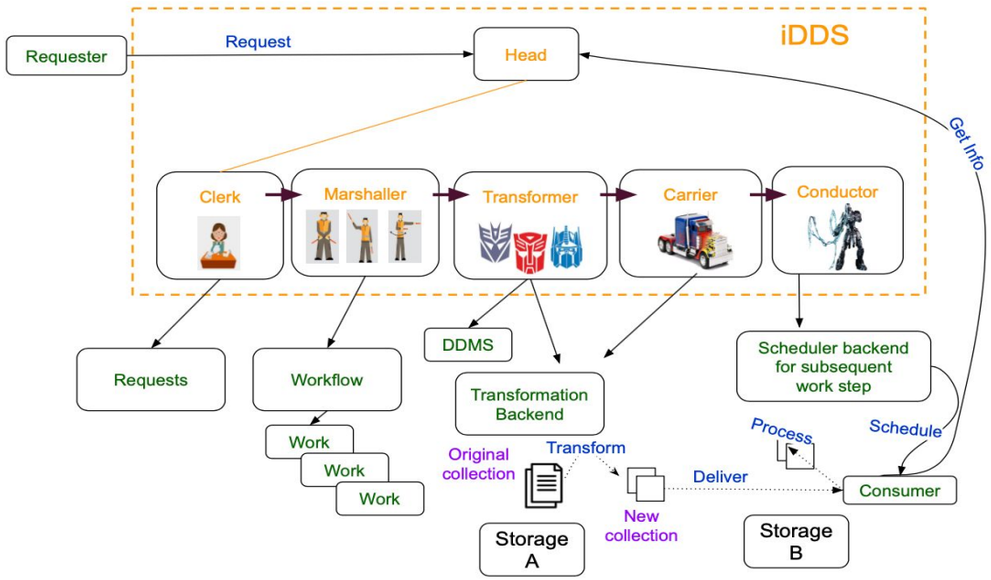
- Distributed heterogeneous computing resources
  - Diverse locations
  - Different software (slurm,condor,pbs)
  - Different work directories and so on
  - Site differences increase user complexity.
- Distributed Users
  - Users from different universities/labs, difficult/impossible to make sure all users have accounts at all sites.
    - LHC 170 sites and several thousands of users
  - Complicated for a users to adapt different software at different sites
- PanDA - Distributed workload Management system
  - General interface for different users at different sites, one authentication for all sites
    - Support X509 and OIDC (CILogon, dexipd and so on) for authentication
  - Integrate different resource providers(Grid, Cloud, Kubernetes, HPC and so on), hide the diversities from users
  - Manage complex workflow, data scheduling, ML and so on.

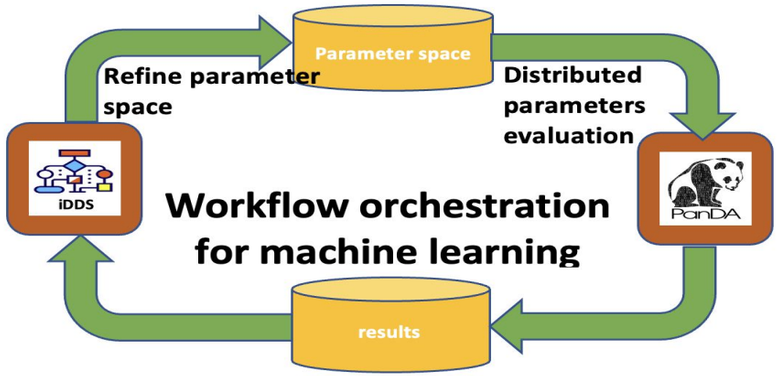
PanDA



- PanDA provides general interface for users
- PanDA and Pilots (red lines) work together as a workload management system to integrate distributed computing resources
  - Pilot works as an agent to acquire the CPUs, validate the environments and pull jobs from PanDA.
  - Pilot starts user jobs and monitors user jobs
- PanDA-pilots works like a distributed virtual cluster.
  - Pilot works as an agent, which can avoid a lot of user job failures.
  - Distributed pilots to hide differences of heterogeneous computing resources.
- Harvester provisions pilots on remote/local resources
- PanDA is more than a virtual cluster
  - Data scheduling (trigger Rucio to move inputs/outputs)
  - Complex workflow management
  - ML and so on.

iDDS (intelligent Data Delivery Service)

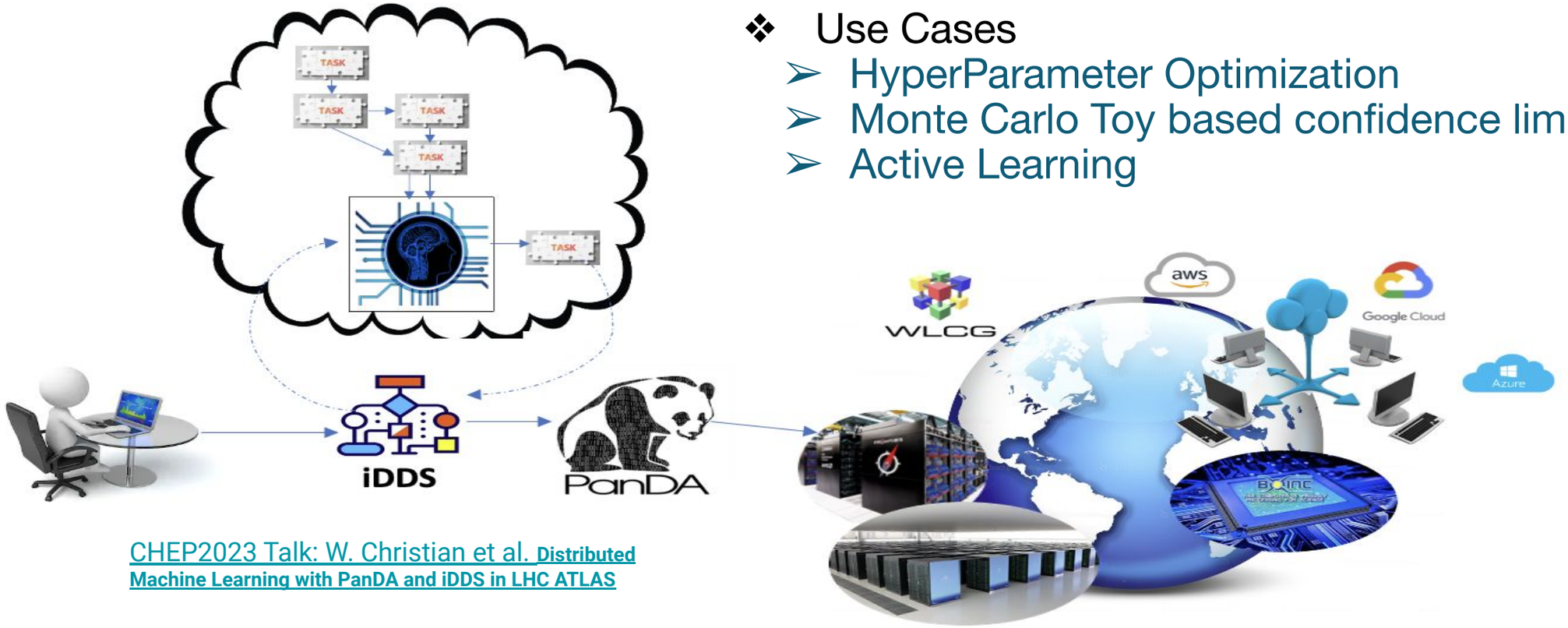




- intelligent Data Delivery Service (iDDS)
  - Intelligent granular data delivery and orchestration supporting complex workflows.
- iDDS orchestrates the workflow for automation
  - Directed Acyclic Graph (DAG), Loops and so on
- Experiment-agnostic service employed by LHC ATLAS, Vera Rubin Observatory
  - Fine-grained Data Carousel for LHC ATLAS enables processing in proper granularities.
  - Rubin DAG workflow management.
  - Scalable ML service to efficiently distribute ML Hyperparameter Optimization tasks and other ML workflows to distributed CPU/GPUS.

Distributed ML with PanDA and iDDS

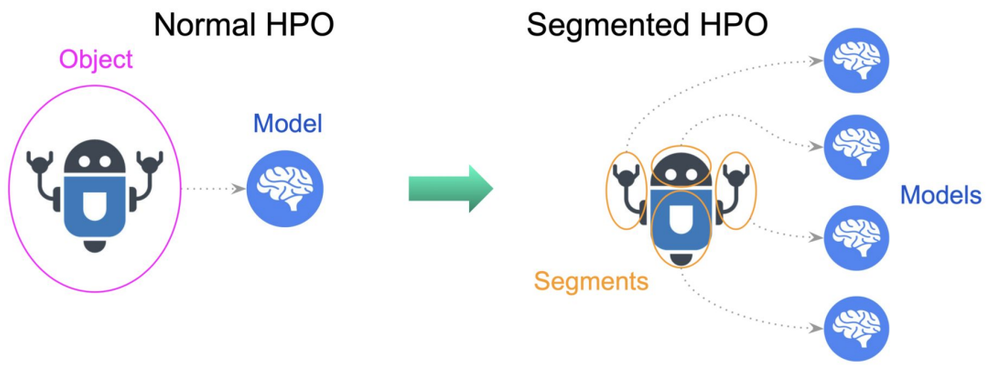
- PanDA as an engine for large scale AI/ML
  - PanDA is powerful to schedule jobs to distributed heterogeneous resources
  - Large scale
  - Transparent to users for different computing resources
  - Smart workload routing
- iDDS (intelligent Data Delivery Service) orchestrates the workflow for automation
  - Complex workflow orchestration
    - Collect results from previous tasks
    - Analyze the results with user predefined jobs
    - Generate new tasks/jobs based on the analyses
- Use Cases
  - HyperParameter Optimization
  - Monte Carlo Toy based confidence limits
  - Active Learning



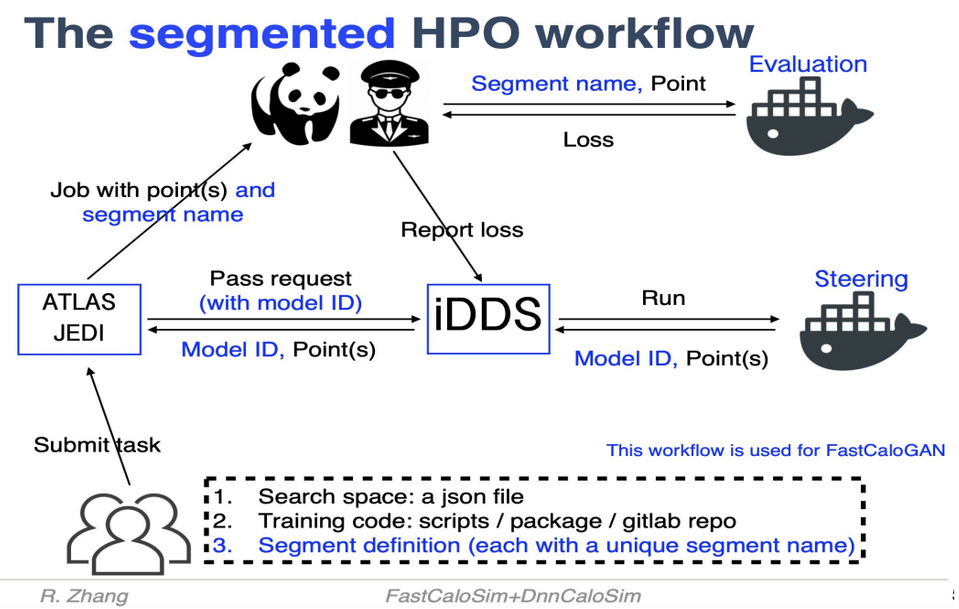
CHEP2023 Talk: W. Christian et al. **Distributed Machine Learning with PanDA and iDDS in LHC ATLAS**

Distributed HyperParameter Optimization (HPO)

- Provide a full-automated platform for HPO on top of distributed computing resources
  - PanDA schedules ML training jobs to distributed heterogeneous GPUs to evaluate the performance of the hyperparameter
  - iDDS orchestrates to collects the results and generate new hyperparameters based on the previous results
- Applied for ATLAS FastCaloGAN
  - The HPO service is in production for FastCaloGAN, part of the production ATLAS fast simulation AtlFast3
  - With hyperparameters to tune various models targeting different particles
  - Distributed GPUs, HPCs, commercial cloud
  - Ref: FastCaloGAN, AML workshop, IML, ATLAS S&C week
- Used in ATLAS, however not specific to ATLAS



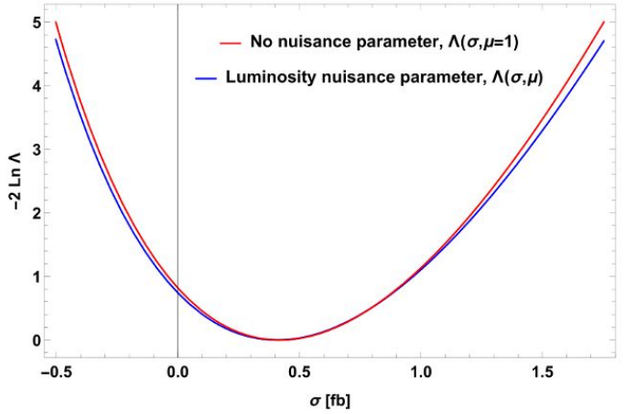
R. Zhang *5th ATLAS Machine Learning Workshop*

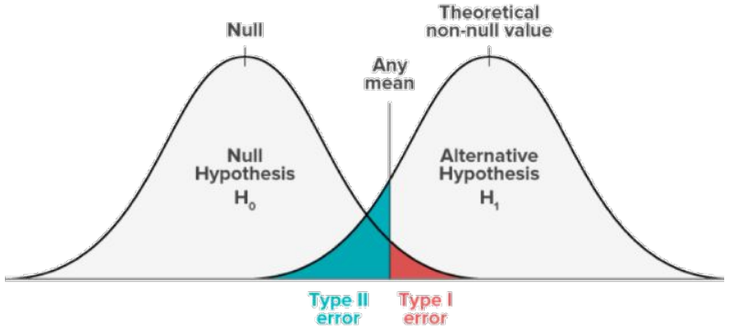


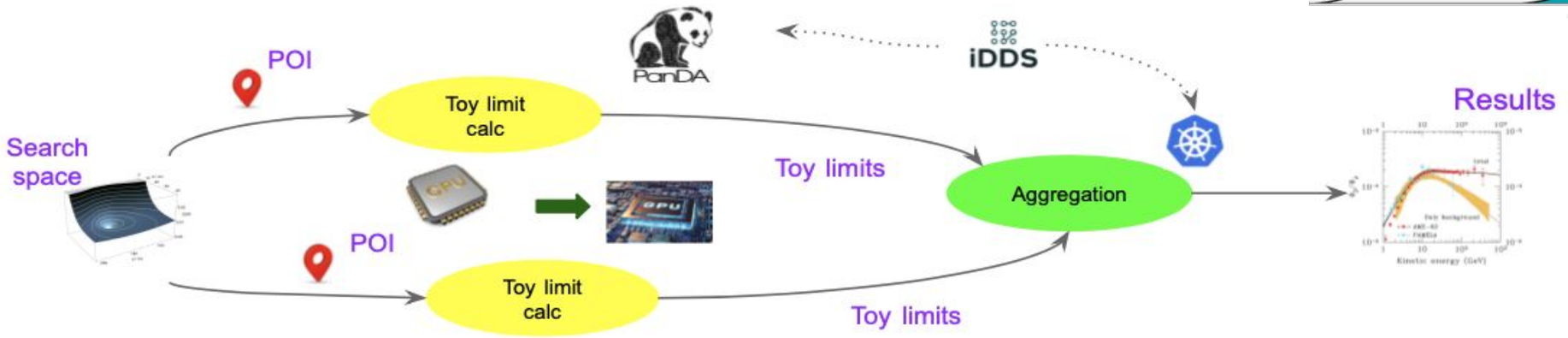
R. Zhang *FastCaloSim+DnnCaloSim*

Monte Carlo Toy based Confidence Limits

- Confidence Limits in Analyses
  - Exclude some ranges of phase space for future processing
  - Show that obtained results are meaningfully different from what could have obtained by chance
- A Monte Carlo (MC) Toy based confidence limits workflow requires multiple steps of grid scans, where the current step depends on the previous steps
- Automate the workflow of Toy limits calculation and aggregation
  - Distributed Toy limits calculation to distributed resources with PanDA
  - Point of Interest (POI) generation based on the search space and results aggregation to generate new POIs in iDDS

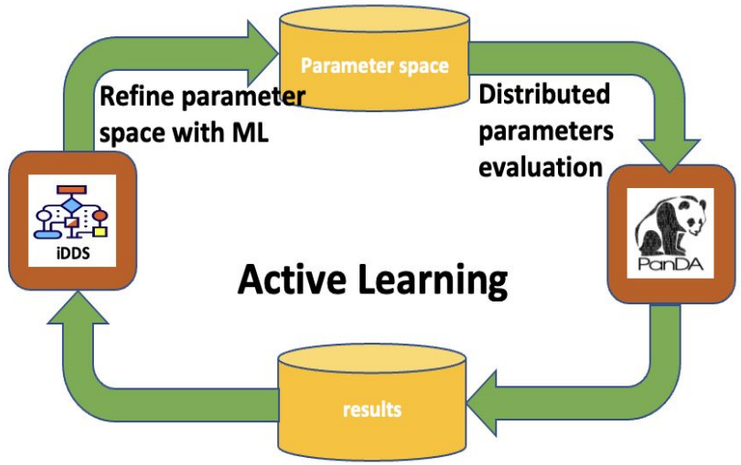


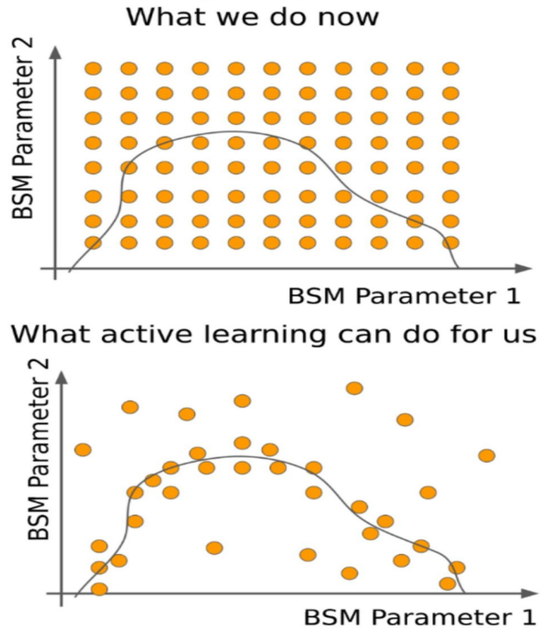




Active Learning

- An iterative ML assisted technique to boost the parameter search in New Physics search space
  - The Active Learning technique we are applying was developed by Kyle Cranmer et al, “Active Learning for Excursion Set Estimation”, ACAT 2019
  - Distributed computing resources for parameter evaluation
  - Optimize the parameter space points for evaluation to maximise the information gain from each evaluation
  - Redefine the parameter space for the next iteration based on the previous results with ML, more efficient than a single-step processing
- Automate the multi-steps processing chain with PanDA and iDDS for ATLAS
  - Integrated REANA (Reusable Analyses) with PanDA for learning processing
  - iDDS orchestrates the workflow to trigger new tasks/jobs based on the previous results

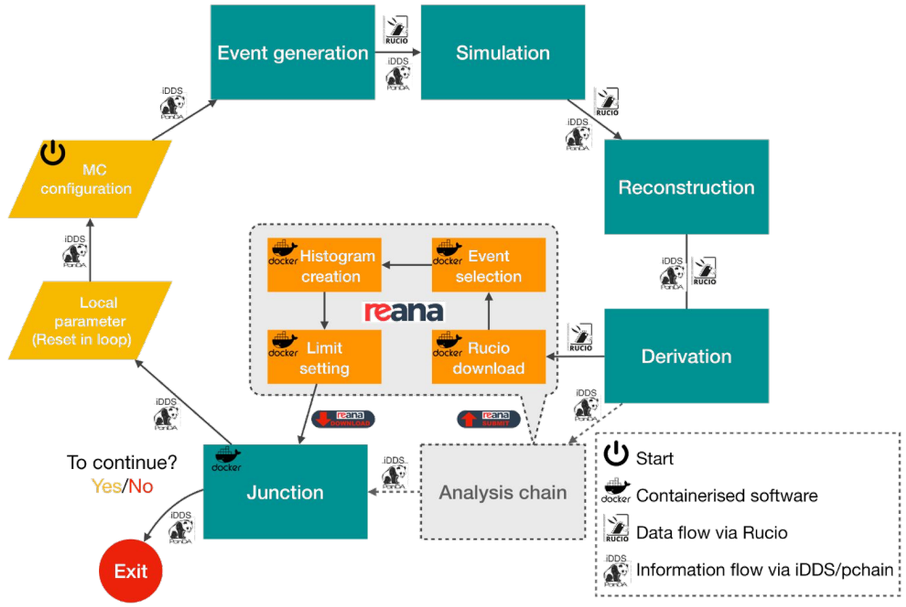


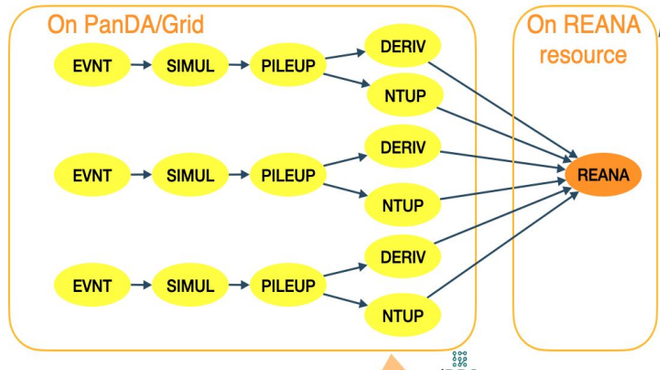


Active Learning via iterative regression on a limit surface

Active Learning for ATLAS

- Applied the Active Learning service in the H $\rightarrow ZZ_d \rightarrow 4\ell$ dark sector analysis
  - Apply Bayesian Optimization to refine the parameter space
  - Greater efficiency, scalability, automation enables a wider parameter search (instead of 1D, 2D or even 4D on large scale resources) and improved physics result
  - Has demonstrated active learning driven re-analysis for dark sector analysis
  - ATLAS PUB Note ATL-PHYS-PUB-2023-010

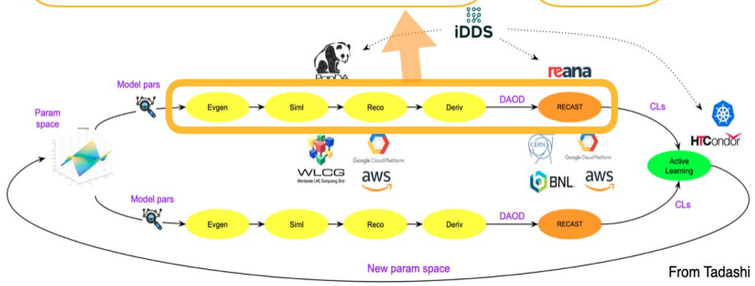




R. Zhang

pchain workflow:

- Three independent chains with different random seeds run in parallel
- Six input files are fed to the final REANA task

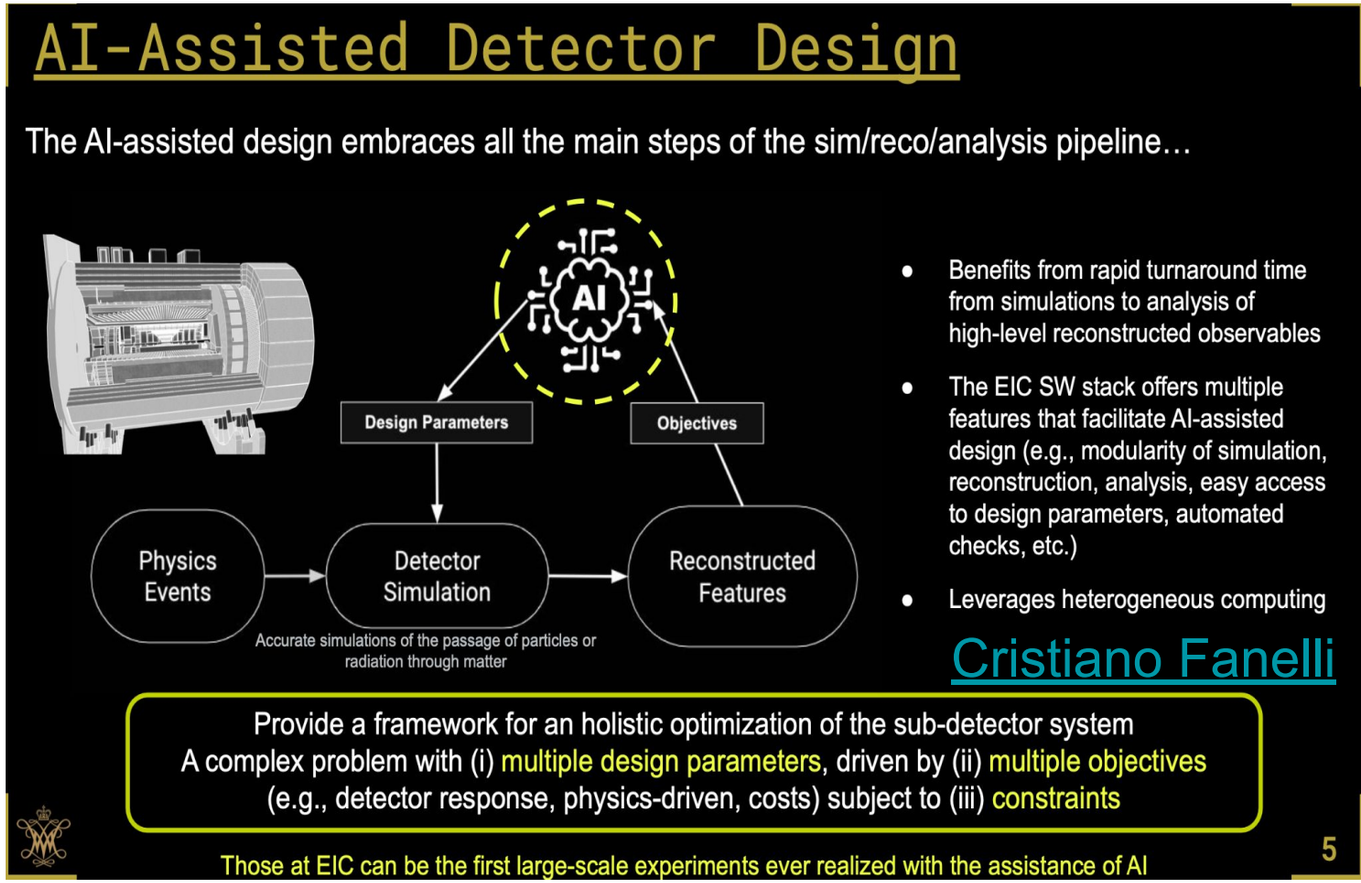


From Tadashi

CHEP2023 Talk: C. Waber, et al. An Active Learning application in a dark matter search with ATLAS PanDA and iDDS

AI-assisted Detector Design at EIC

- A 2 year project supported by DOE NP
- Collaborators
  - W&M,BNL,JLab,Duke,CUA
- Scalable distributed AI-driven Detector Design
- Develop a modular, customizable, scalable pipeline
- Here we will only focus on a part of the project, to adapt/develop a scalable AI/ML workflow with PanDA/iDDS
  - AI-driven scalable distributed parameter optimization



Distributed Optimization in AID(2)E

- AI-driven scalable & distributed parameter optimization
  - Many parameters with big search space, many concurrent jobs are required
    - Scalable and distributed resource with PanDA
  - Multiple objectives and multiple-steps optimization
    - Multiple objective optimization
    - Multiple objective bayesian optimization
    - Orchestration based on previous results with multiple-steps
    - Workflow orchestration with iDDS

How do we design and optimize detectors?

- Typically full detector design is studied once the subsystem prototypes are ready.
- In the subsystem design phase constraints from the full detector or outer layers are taken into consideration.
- Actually many parameters (mechanics, geometry, optics) characterize the design of each sub-detector, hence the full design represents a large combinatorial problem. A well known phenomenon observed in optimization problems with high-dimensional spaces is the so-called "curse of dimensionality" [1], introduced for the first time by Bellman when considering problems in dynamic programming.
- In addition to that, more objective functions often need to be considered at the same time in the design of each sub-detector (e.g., resolution, efficiency, cost, distinguishing power, etc).
- In this context, AI offers SOTA solutions to solve complex optimization problems in an efficient way.

Cristiano Fanelli

[1] Bellman, Richard. *Dynamic programming*. Vol. 2[illegible] RAND CORP SANTA MONICA CA, 1956.

8

Developments undergoing in PanDA/iDDS

- Optimize the user interface to adapt AID(2)E
  - Python decorator based workflow definition and processing
- Async result handling
  - Messaging based result processing
  - To efficiently and seamlessly transfer inputs and outputs between different jobs in a very large scale
  - Current ask-tell mode in iDDS HyperParameter Optimization does not fit
- Logs
  - Basic log handling based on PanDA
  - If possible, realtime logging will helpful

Near term plans

- Environment customization
  - PanDA/iDDS environment setup and customization
  - CILogon based on authentication/authorization for different collaborators
  - Computing resource customization
    - Where to run jobs
    - How to send pilots to different locations
    - How to run jobs: environment setup (singularity) and so on
- Workflow structure development and adaptation
  - Optimize user interface to adapt AID(2)E
  - Adapt to AID(2)E tutorial/hello-world workflow
  - Adapt to AID(2)E workflow
- Integration tests
- AI/ML method optimization

Thanks