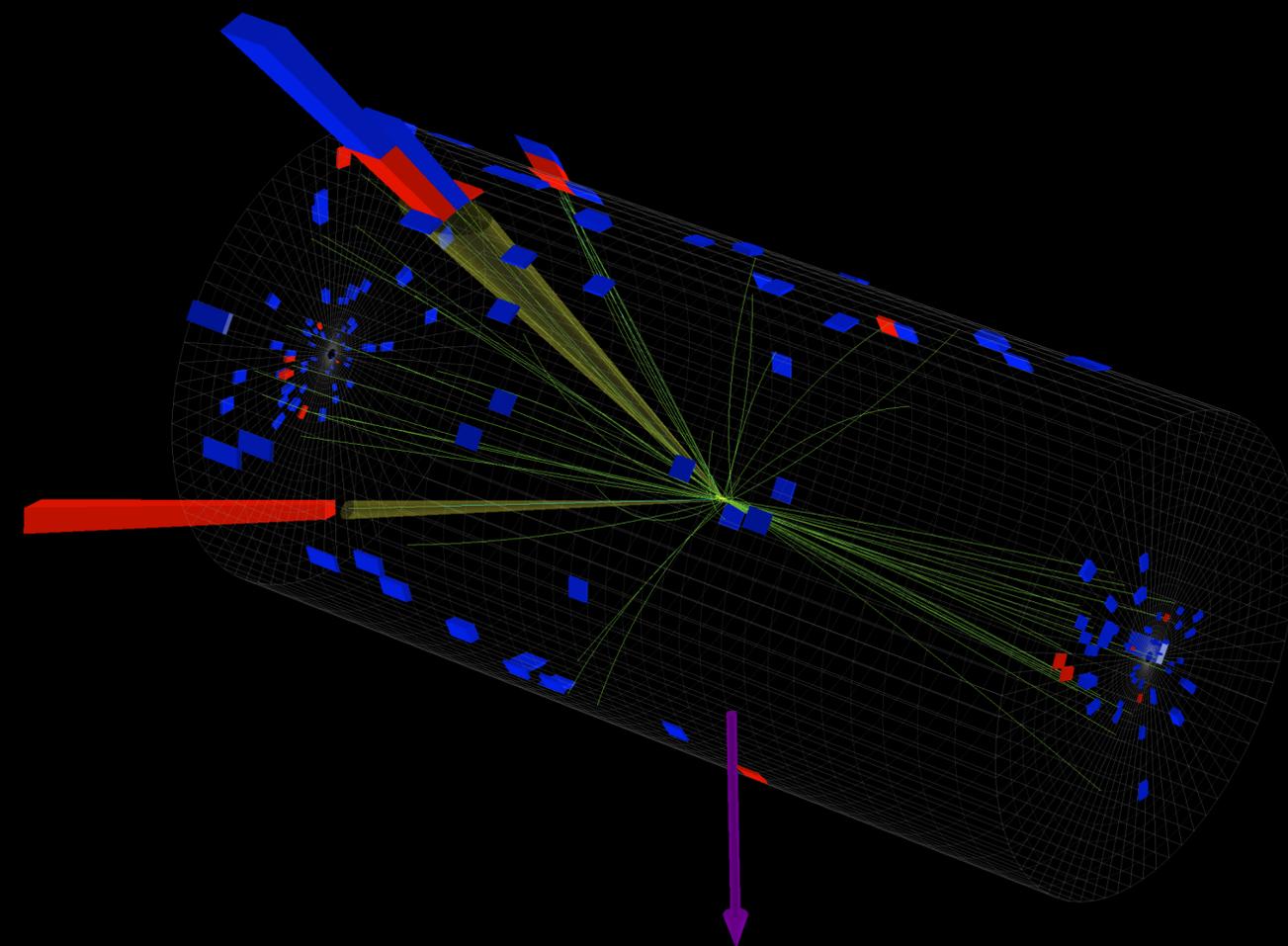


ML for Jets and Beyond Jets

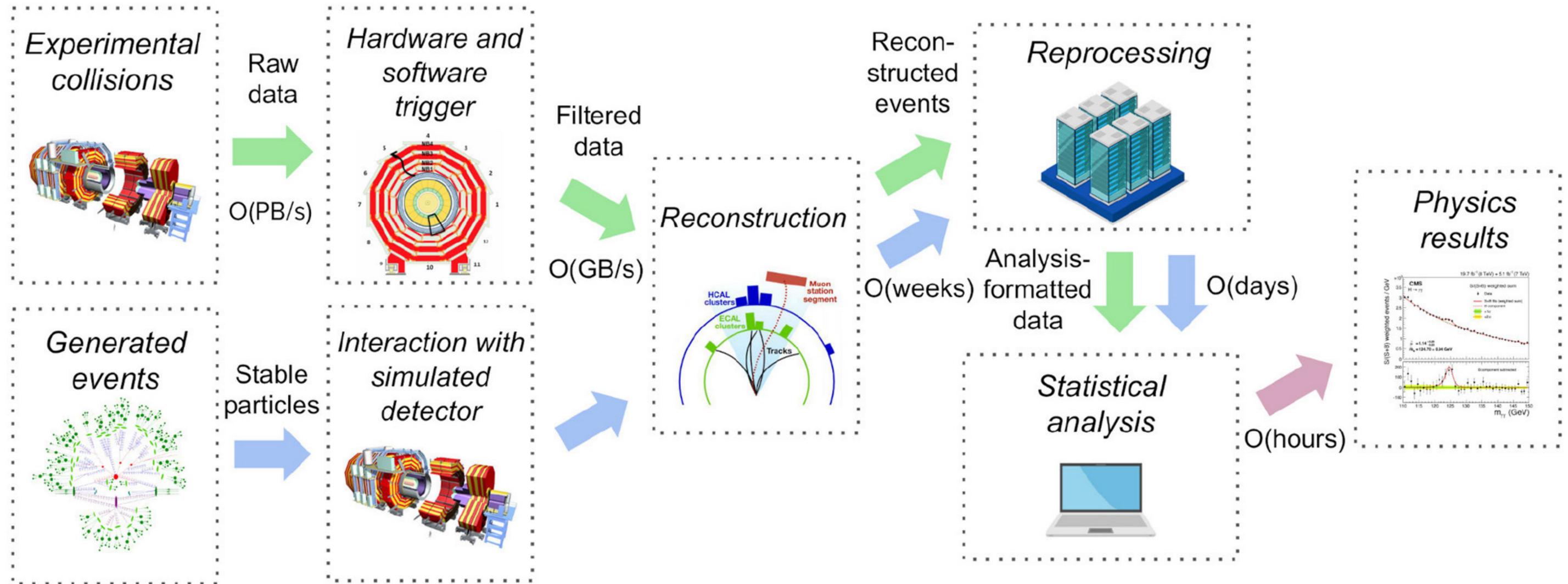
Huilin Qu

ML4jets2023

08.11.2023

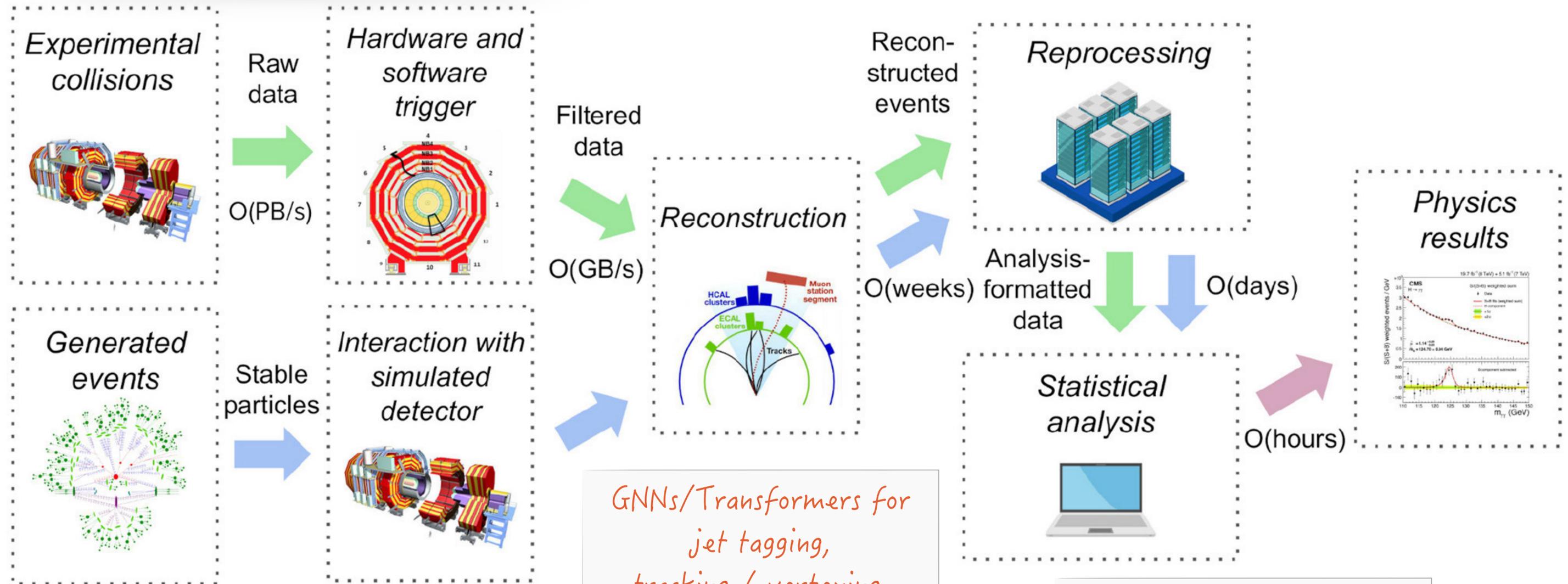


THE HEP-EX FLOW...



THE HEP-EX FLOW... MATCHED WITH ML!

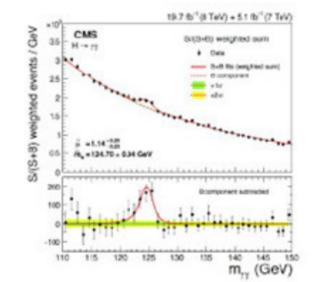
Ultrafast inference (FPGA/ASIC),
anomaly detection...



Generative models for
event generation & fast simulation:
GAN, VAE, normalizing flow,
diffusion...

GNNs/Transformers for
jet tagging,
tracking / vertexing,
particle-flow,
calorimeter reconstruction...

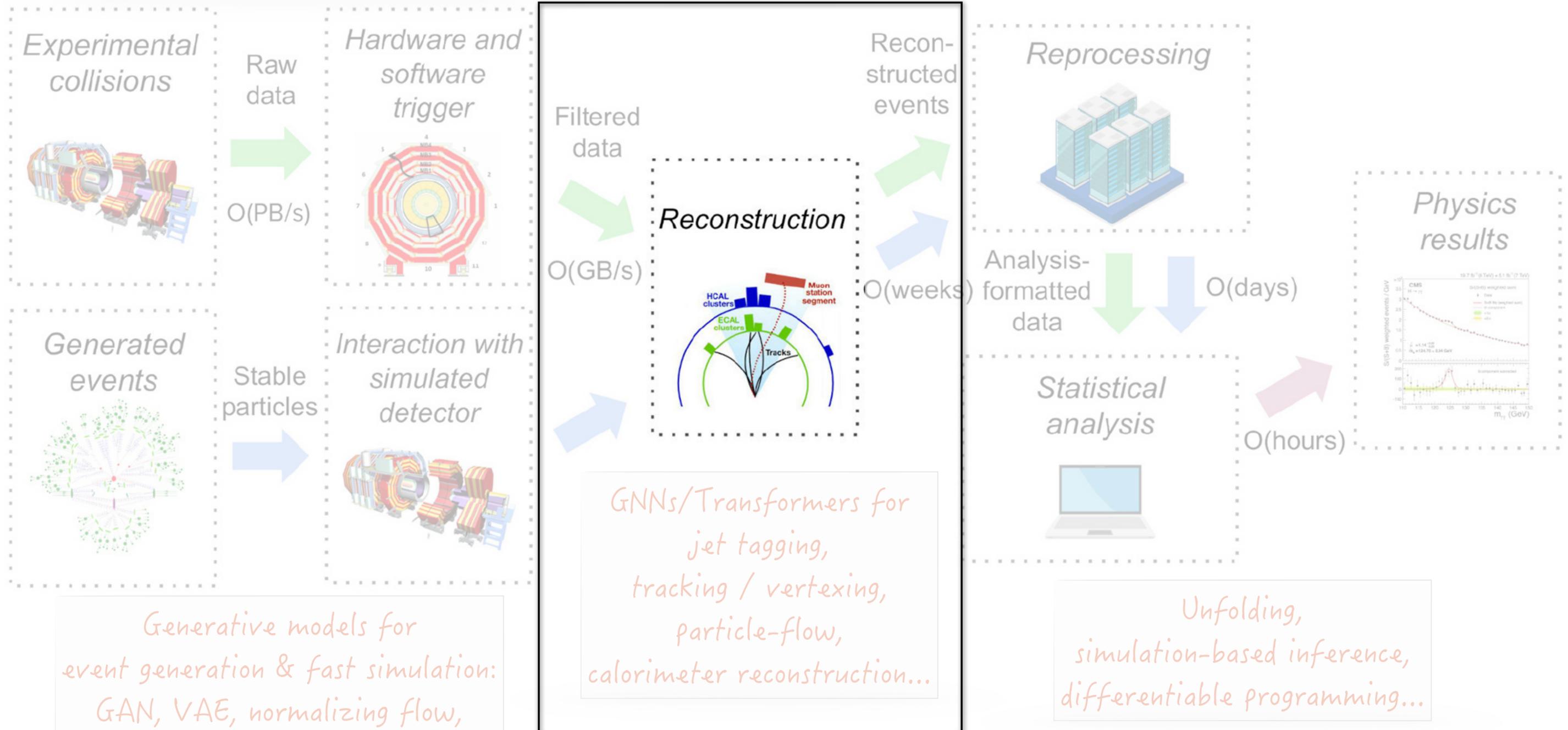
Unfolding,
simulation-based inference,
differentiable programming...



THE HEP-EX FLOW... MATCHED WITH ML!

Ultrafast inference (FPGA/ASIC),
anomaly detection...

Focus of this talk



Disclaimer

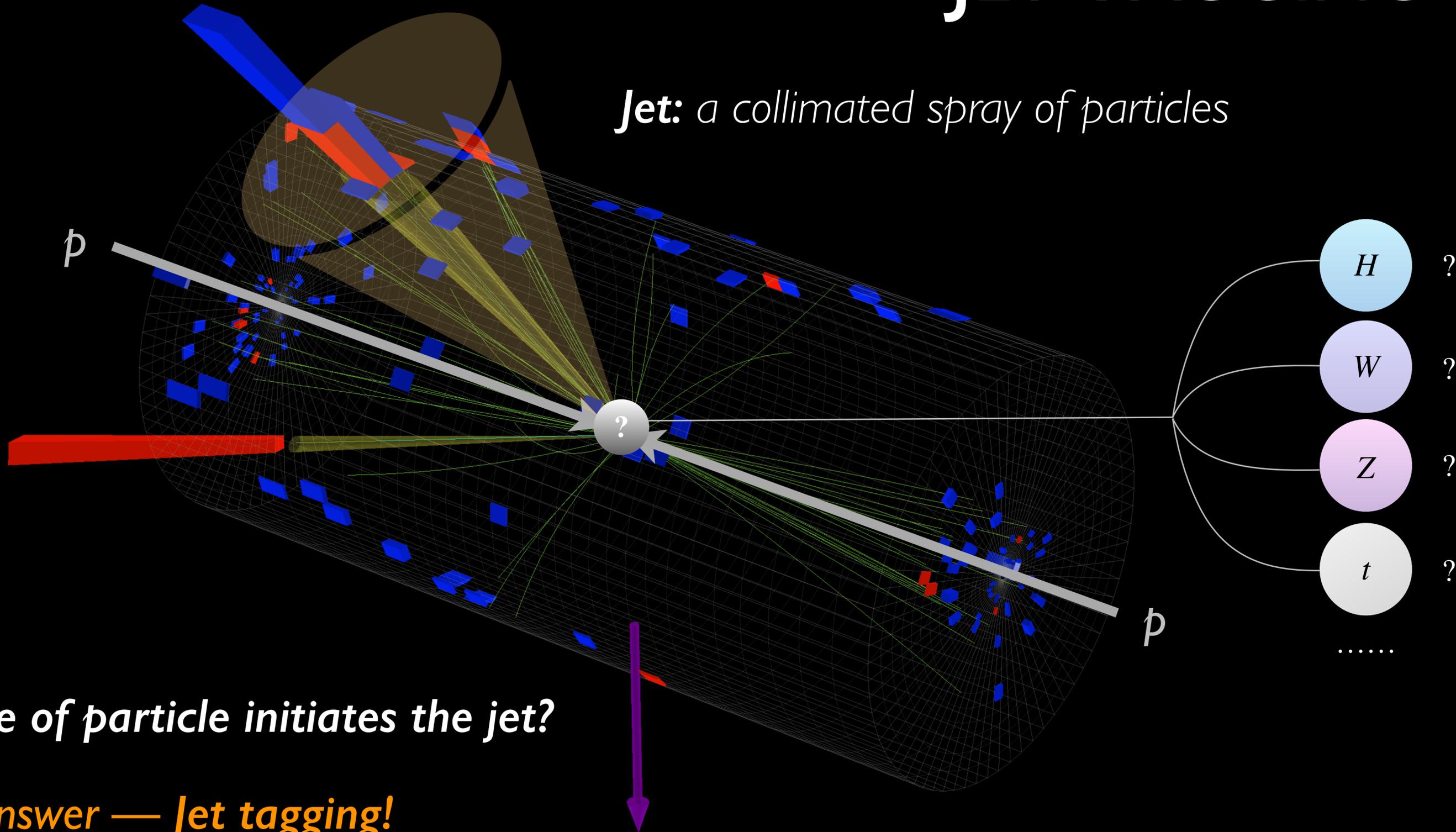
A very personal perspective—
Sorry if your favorite topics/ideas are missing...
...and please enlighten me!



CMS Experiment at LHC, CERN
Data recorded: Sat Aug 5 15:32:22 2017 CEST
Run/Event: 300515 / 205888132

JET TAGGING

Jet: a collimated spray of particles



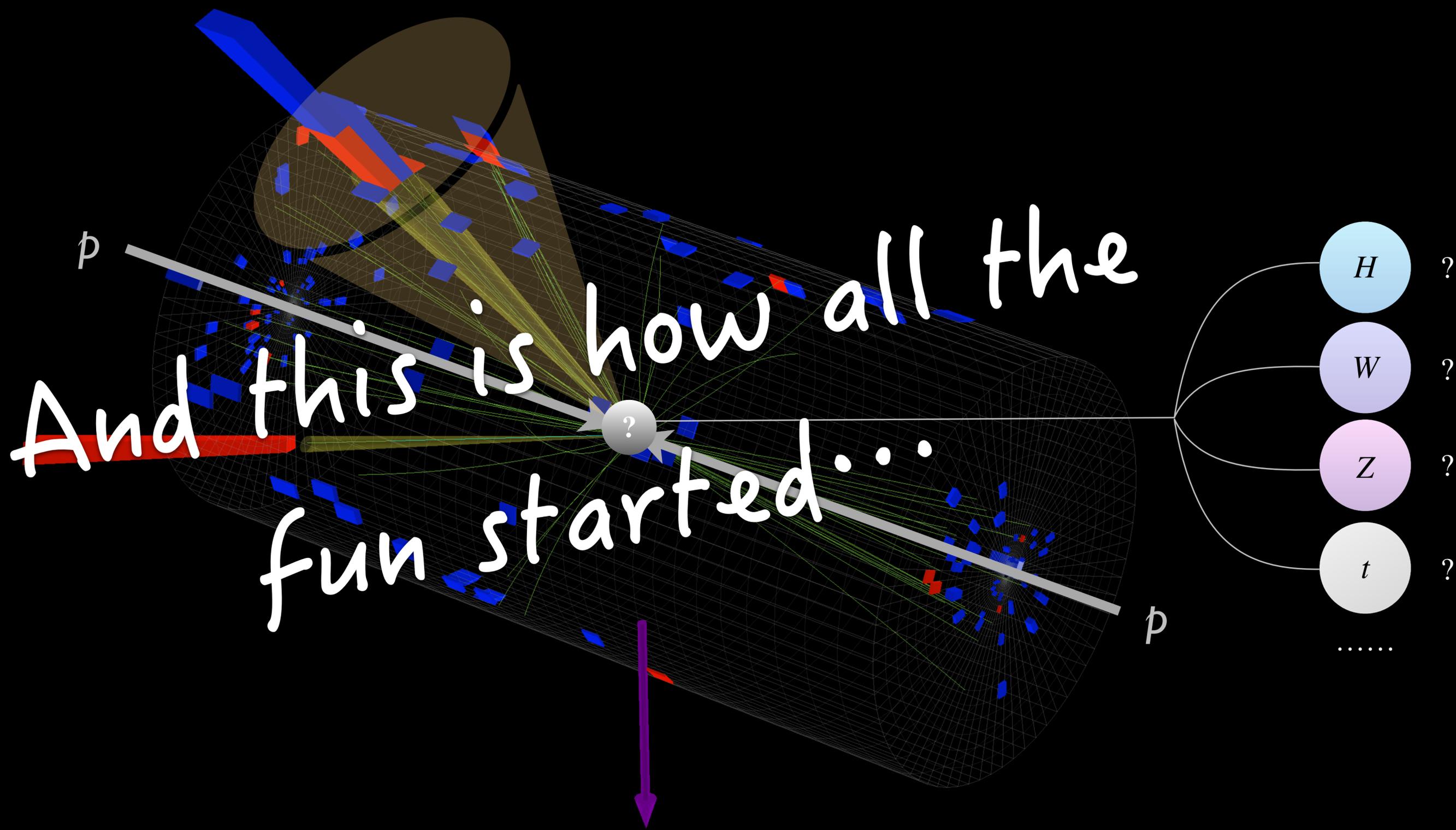
Key question:

What type of particle initiates the jet?

The answer — Jet tagging!



CMS Experiment at LHC, CERN
Data recorded: Sat Aug 5 15:32:22 2017 CEST
Run/Event: 300515 / 205888132



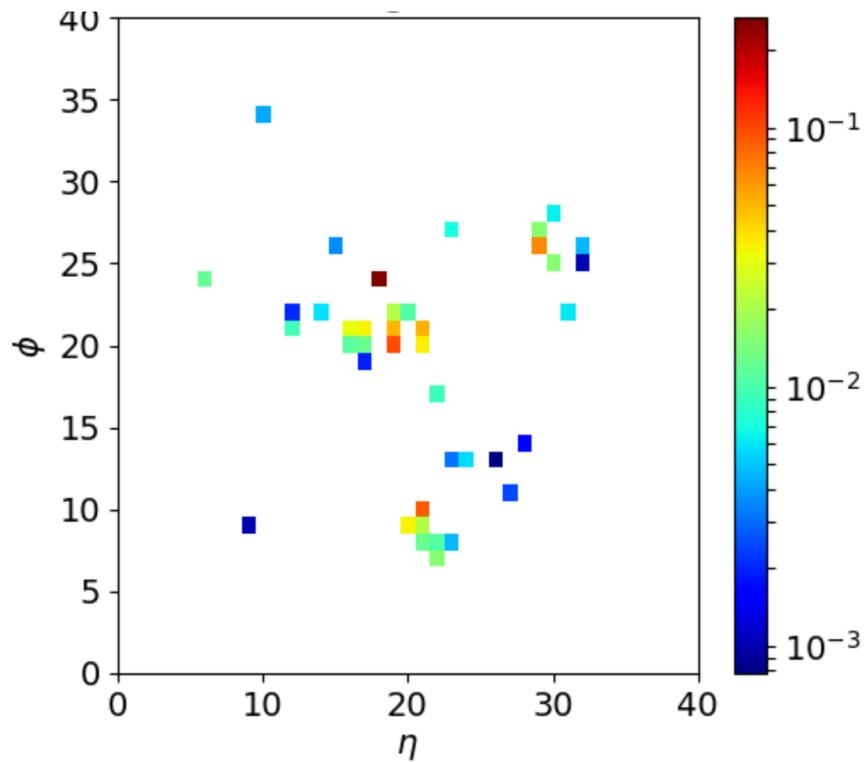
And this is how all the fun started...

- H ?
- W ?
- Z ?
- t ?
-

A BIT OF HISTORY...

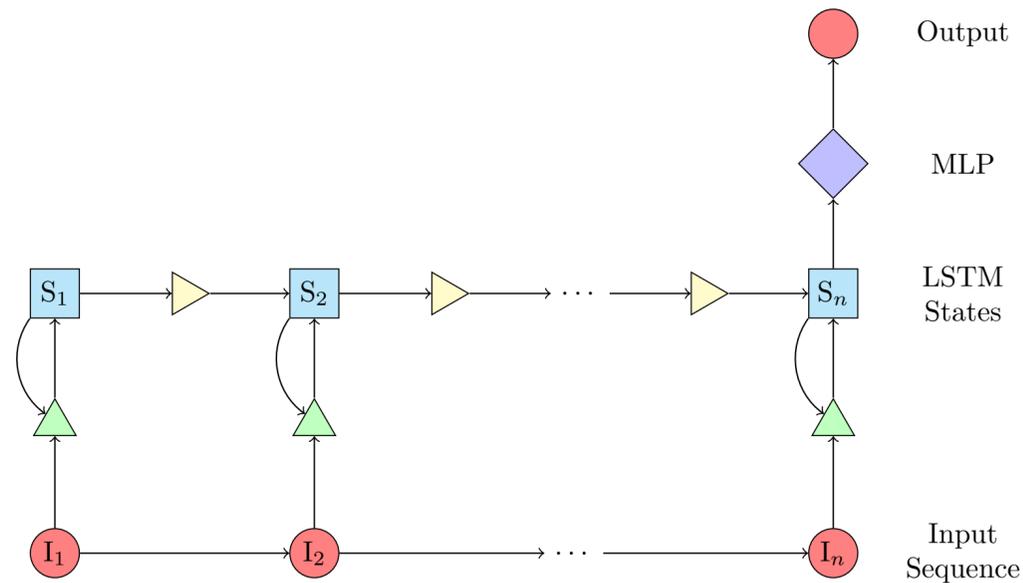
Image

+ 2D ConvNets, ...



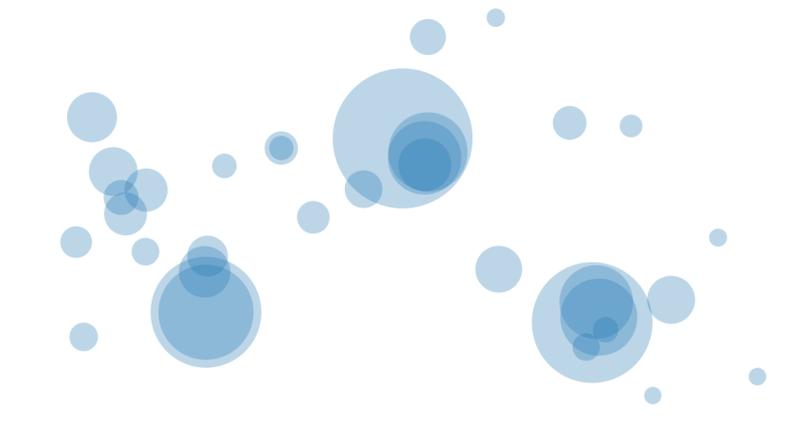
Sequence

+ RNNs, 1D ConvNets, ...



Point / Particle cloud

+ DeepSets, GNNs, Transformers, ...



2015

2016

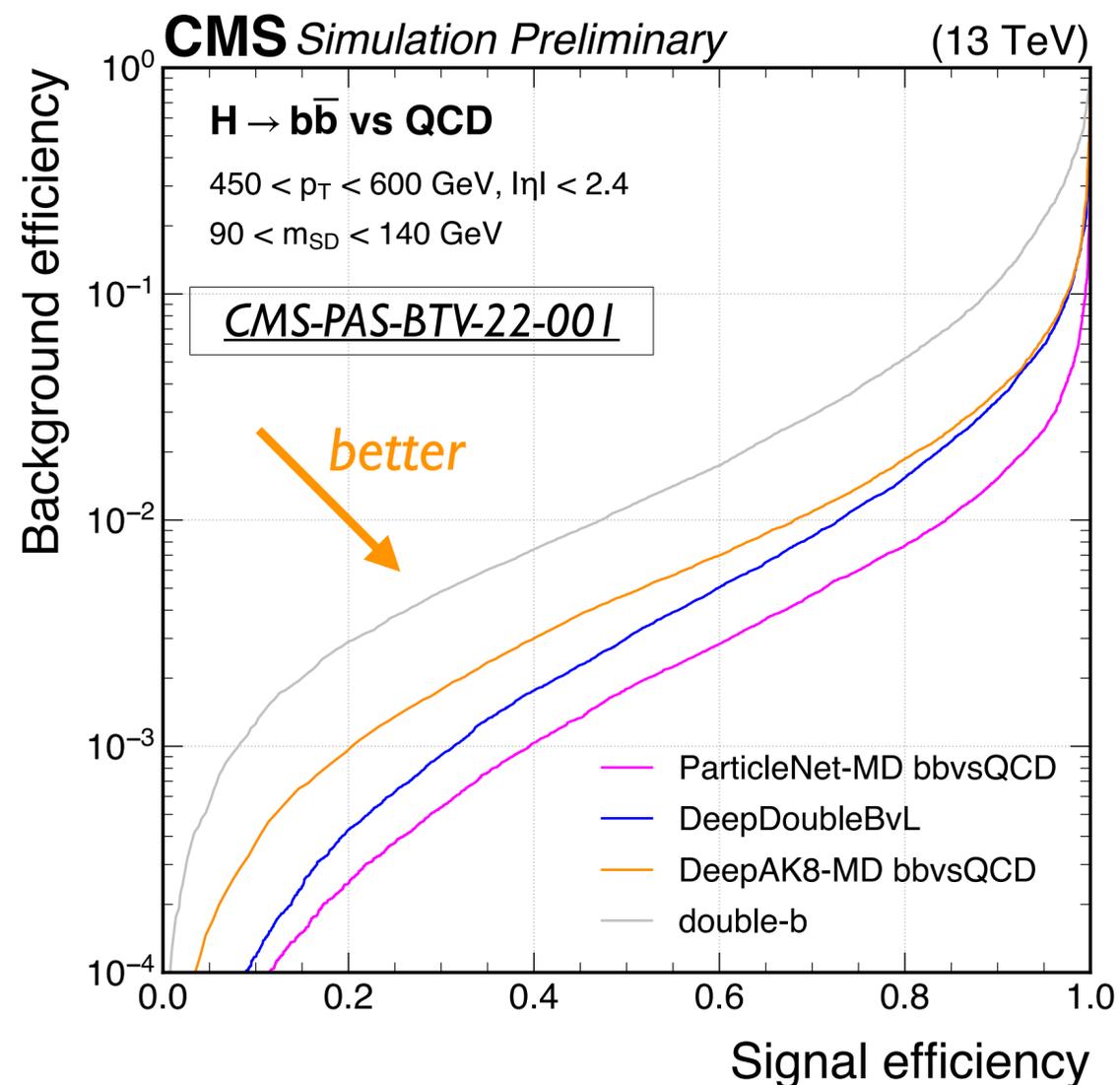
2017

2018

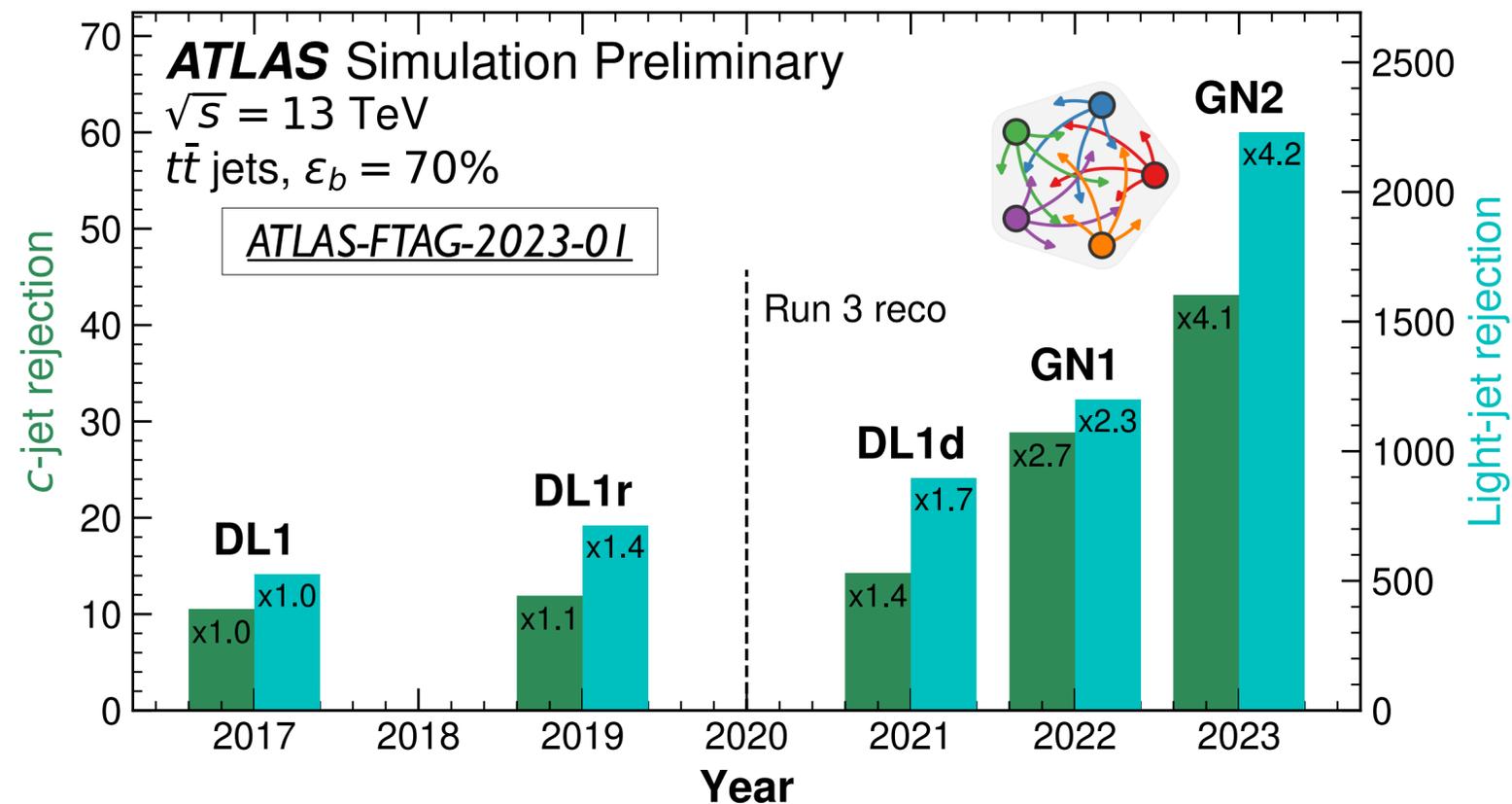
JET TAGGING IN ACTION

- GNN/Transformer-based taggers have become the standard in ATLAS & CMS
- leading to significant performance improvement in recent years

H → *bb* tagging with large-R jets



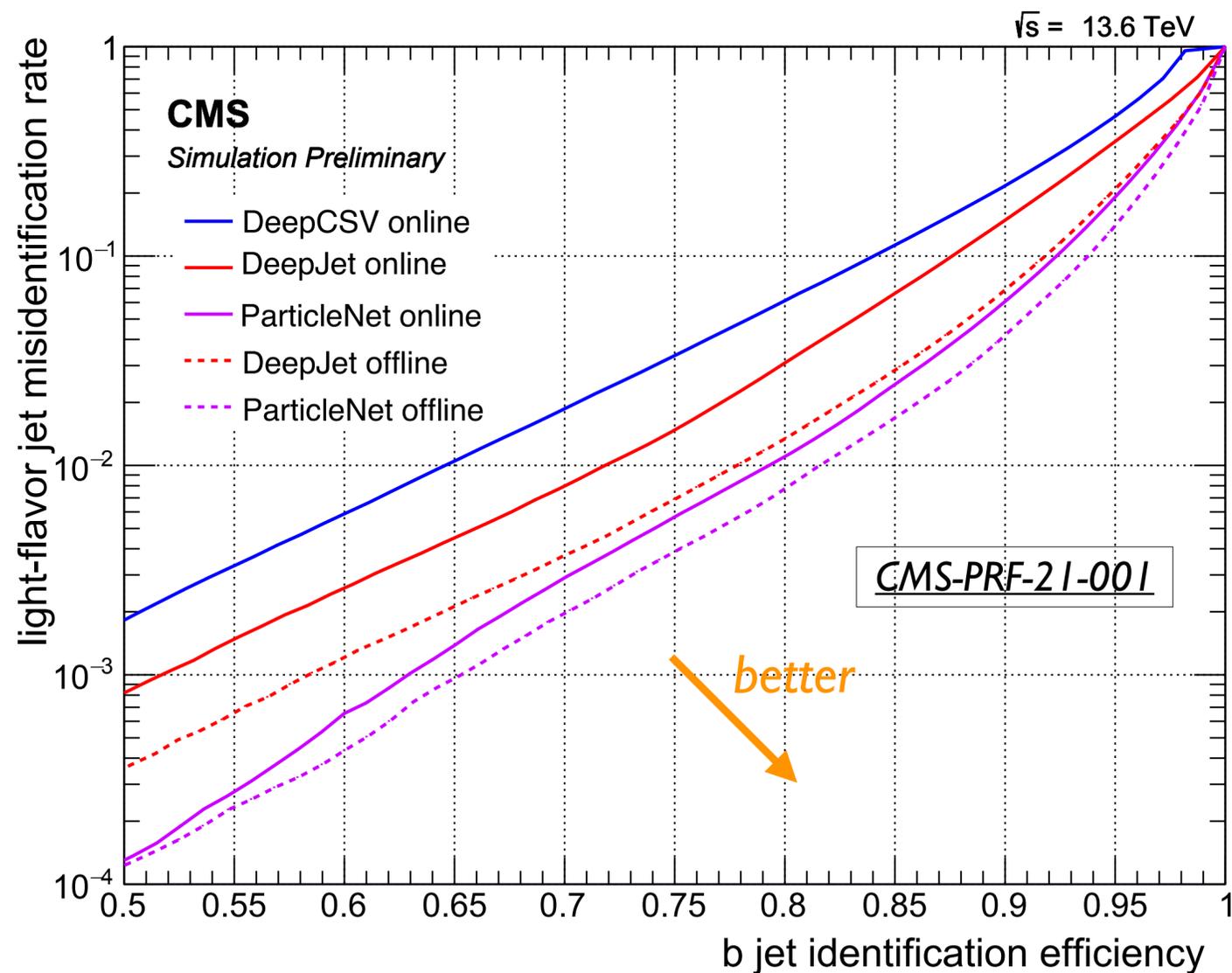
Jet flavor tagging with small-R jets



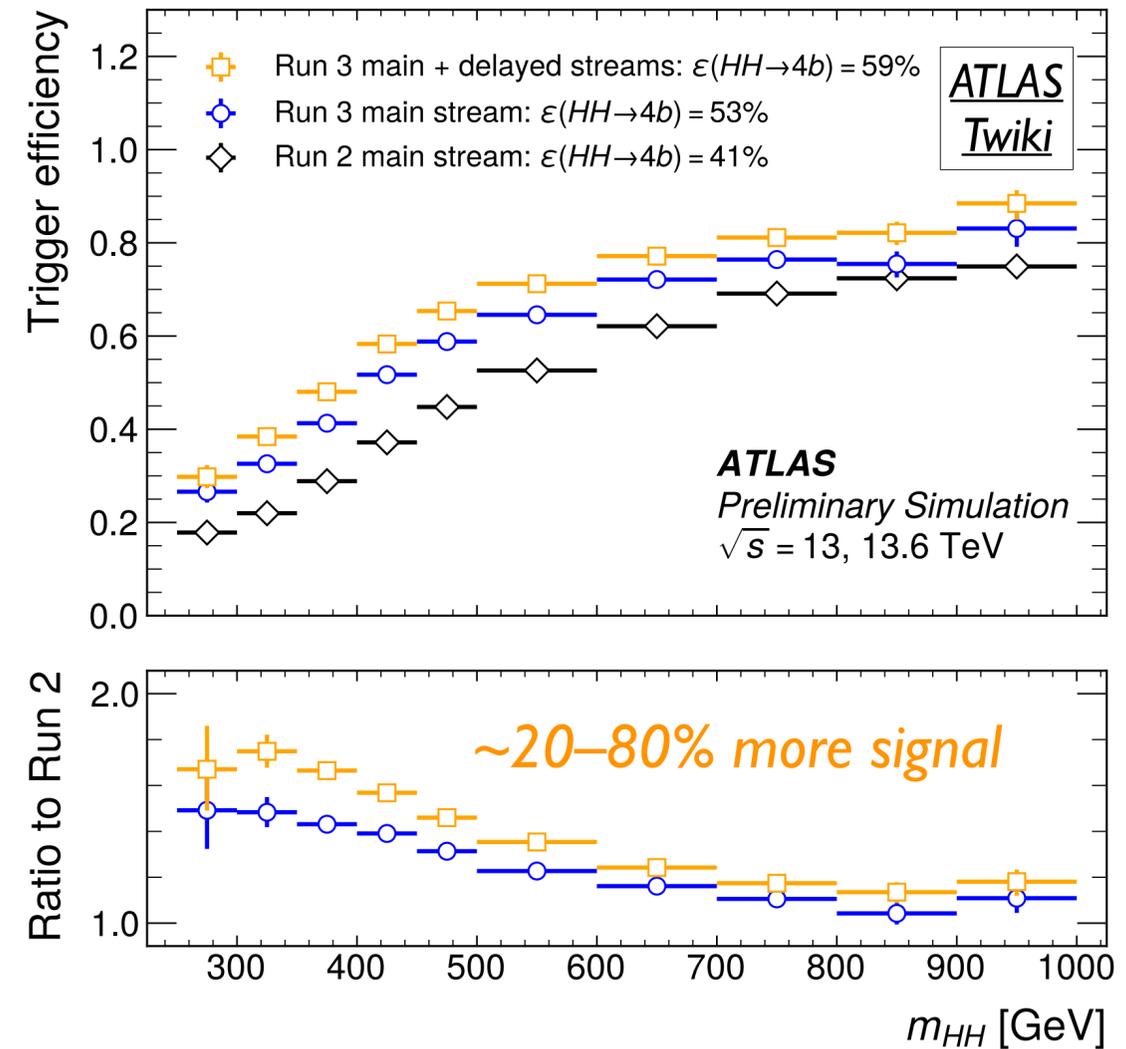
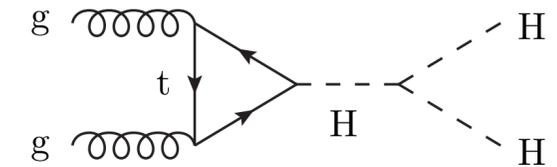
JET TAGGING FOR ONLINE EVENT SELECTION

- These state-of-the-art taggers also deployed at the High-Level Trigger (HLT) system for online event selection
 - substantial improvement in trigger efficiency for e.g., di-Higgs searches

b-tagging performance @ HLT

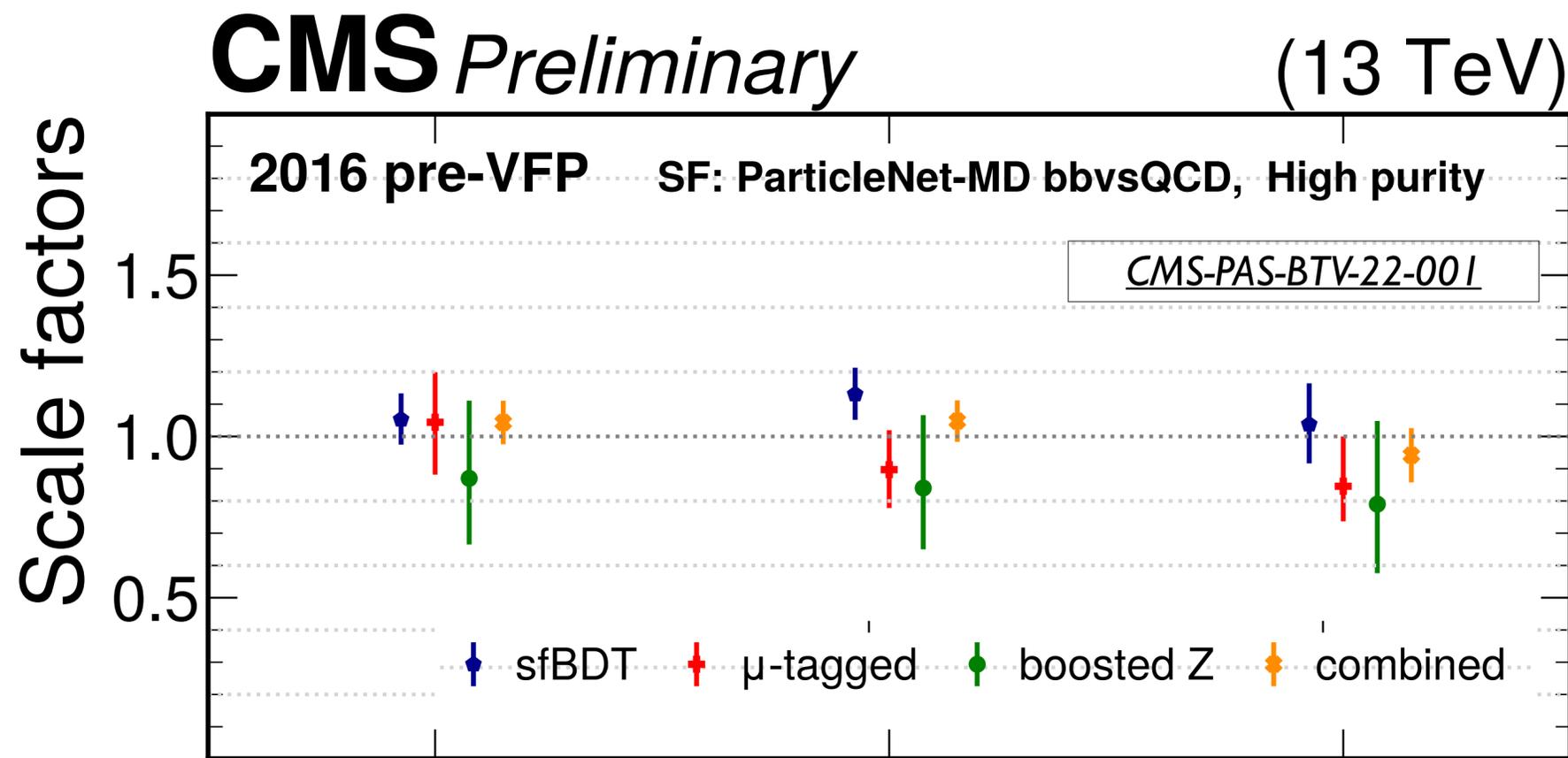


Trigger efficiency for $HH \rightarrow 4b$

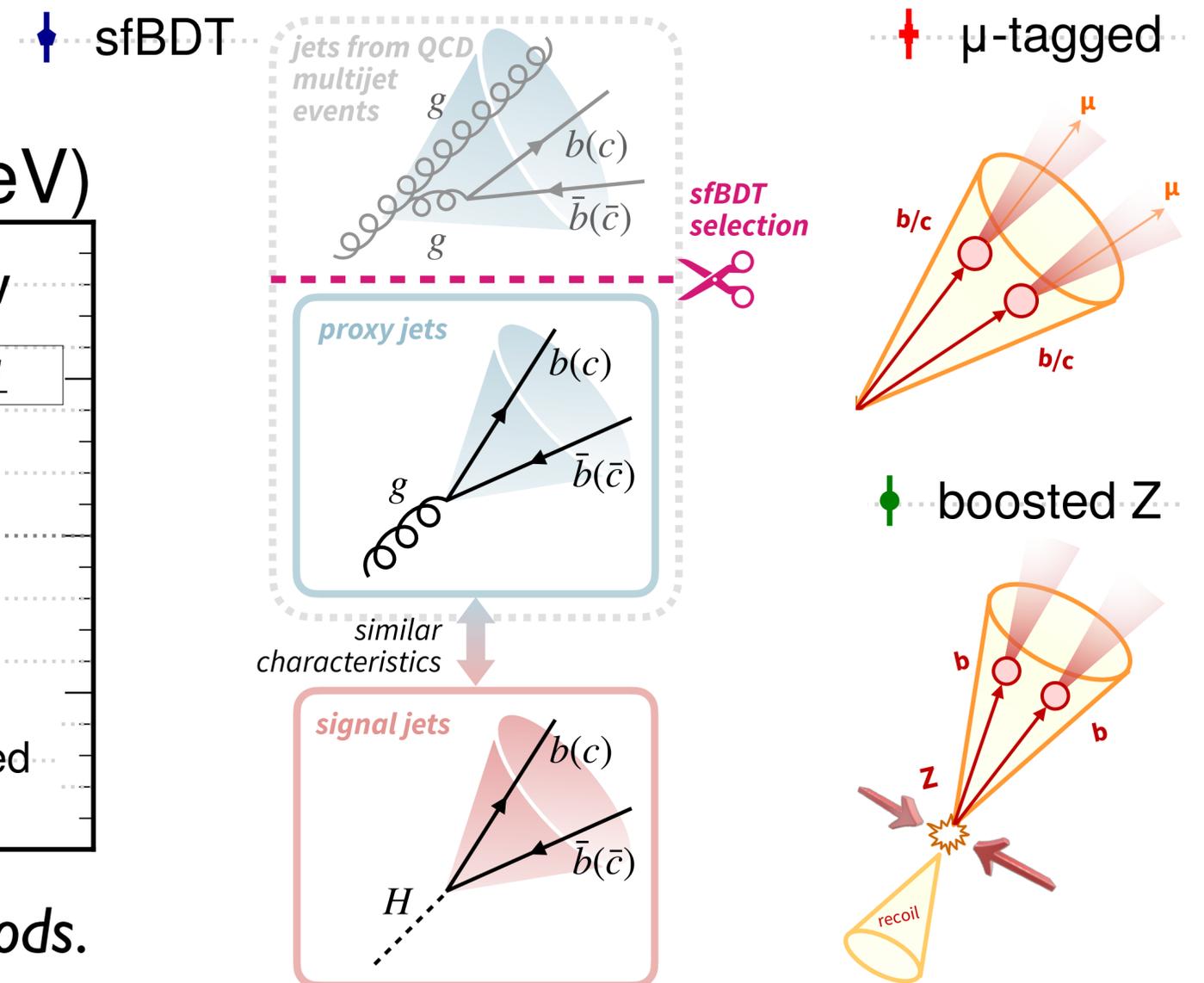


CHALLENGES FOR CALIBRATION

- More powerful taggers bring also more challenges to the calibration
 - calibration often relies on “proxy” jets similar to signal: e.g., gluon \rightarrow bb as a proxy for $H \rightarrow$ bb
 - but the proxy \leftrightarrow signal similarity can break down as the ML taggers get more and more powerful
 - more care and innovations needed in defining proxy jets



Consistent results between different methods.

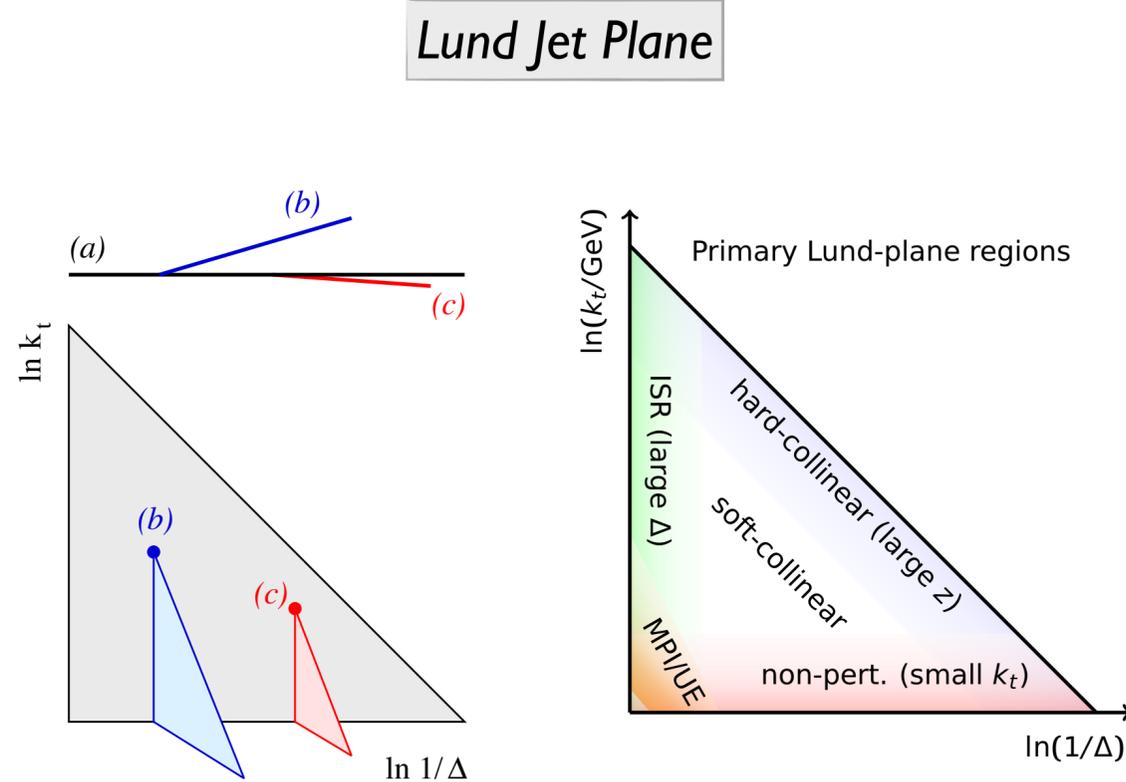


CHALLENGES FOR CALIBRATION (II)

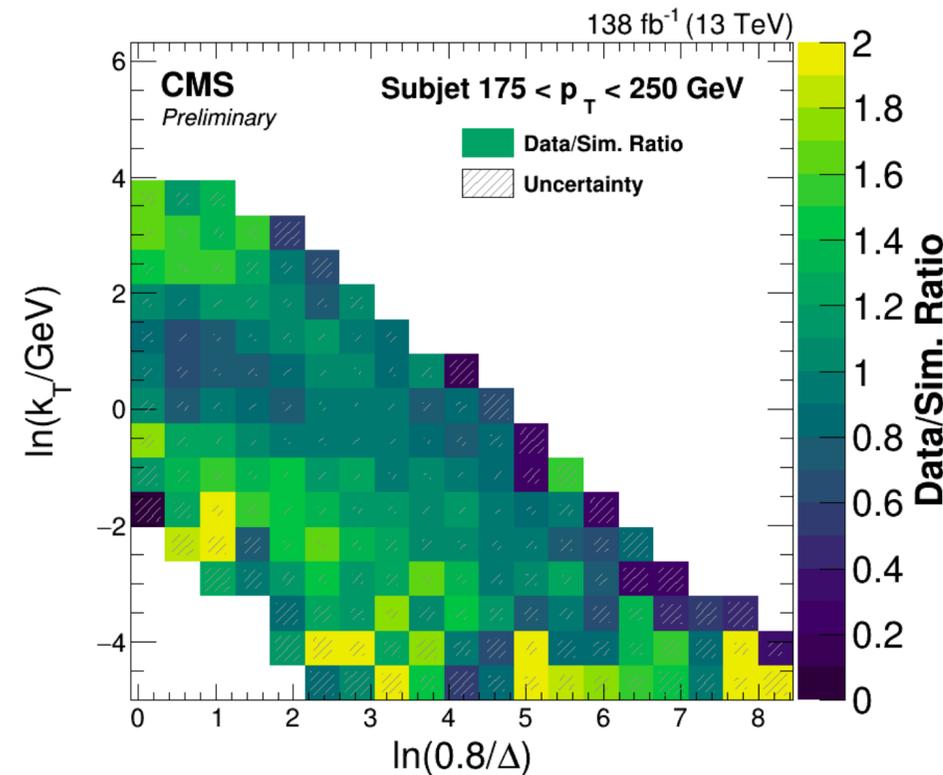
- But there are signals for which no suitable proxy exists
 - e.g., $H \rightarrow WW^* \rightarrow 4q$, multi-prong jets from new physics, ...
- New approach: Lund Jet Plane (LJP) Reweighting
 - recluster AK8 jet exclusively into N subjets (e.g., $N=4$ for $H \rightarrow WW^* \rightarrow 4q$)
 - correct the radiation patterns of each subjet using data/MC ratios of the LJP density
 - LJP density ratios measured using $W \rightarrow qq$ subjets from semileptonic $t\bar{t}$ events

CMS-DP-2023-046

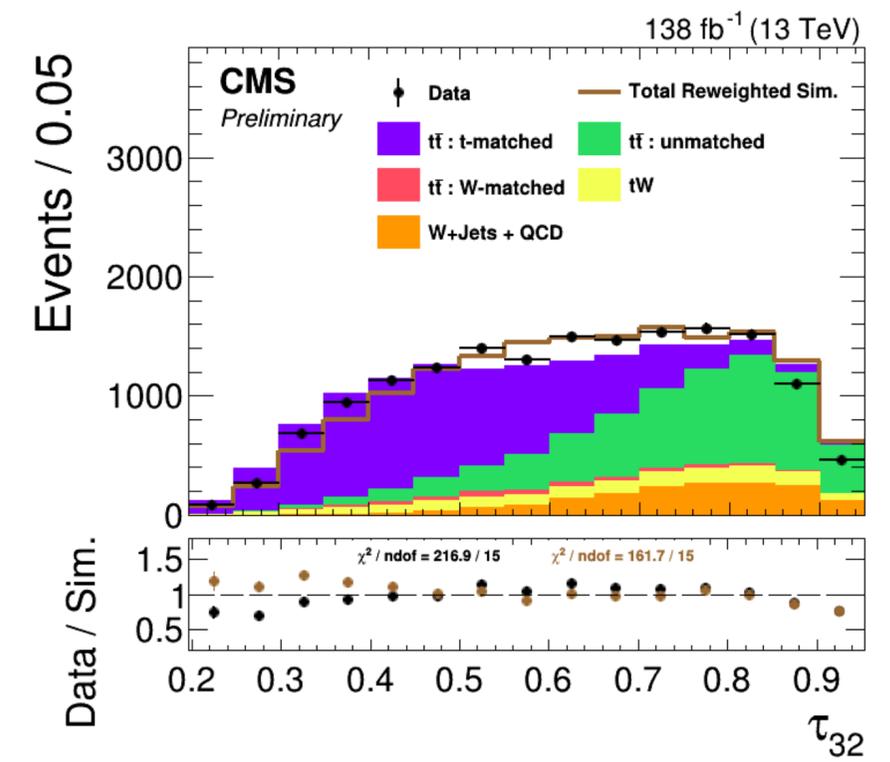
JET
LUND DIAGRAM



LJP ratios

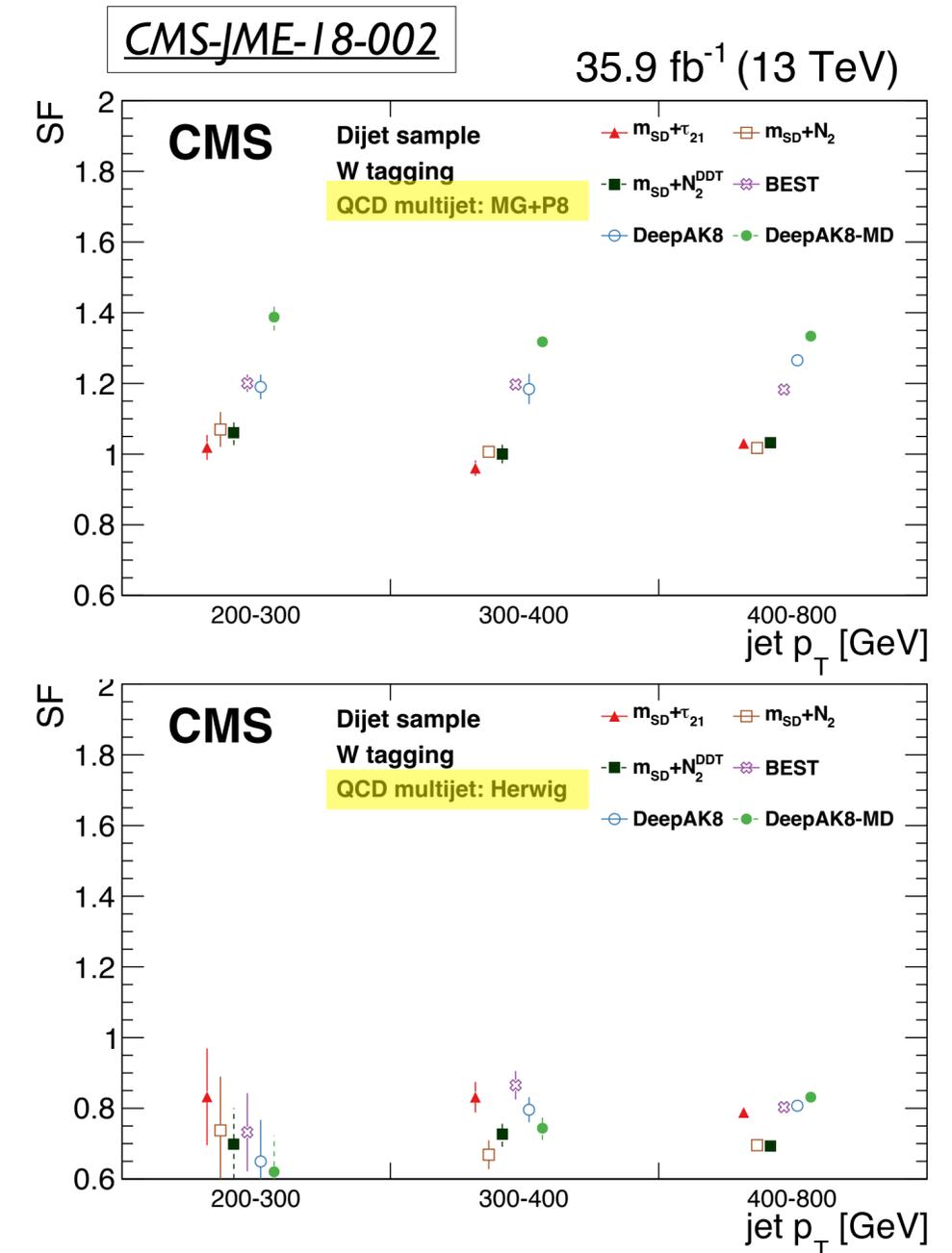
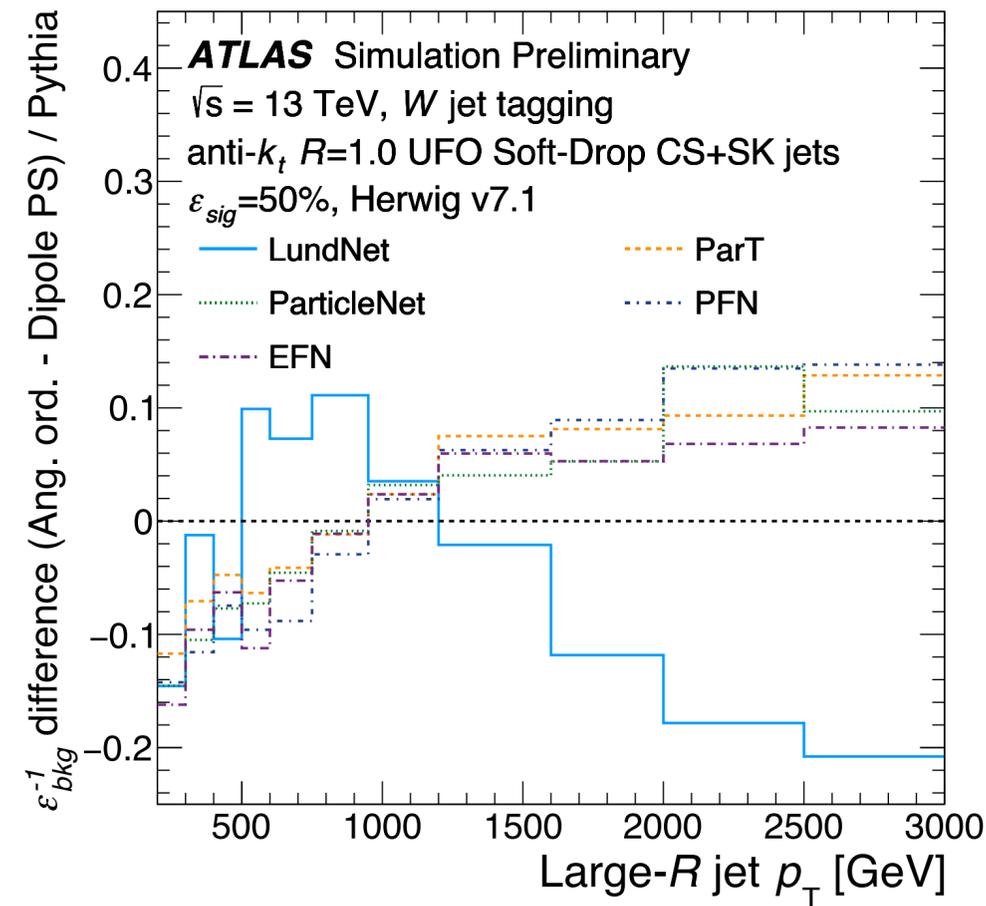
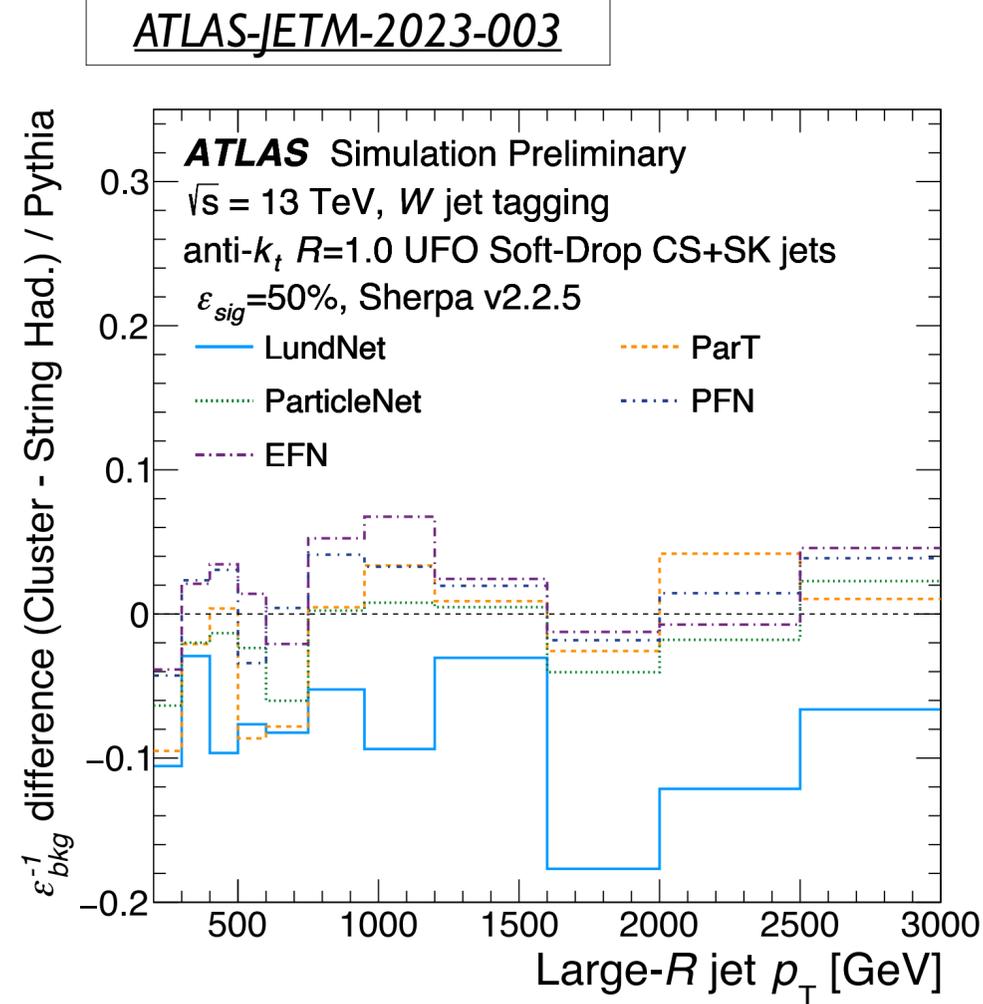


LJP reweighting for 3-prong top jets



SYSTEMATIC UNCERTAINTIES

- More powerful taggers appear to be more sensitive to the modeling of simulation
 - e.g., large variations in background misidentification rates between different parton shower/hadronization models
 - => potentially larger systematic uncertainties...



DESIGNING ROBUST TAGGERS

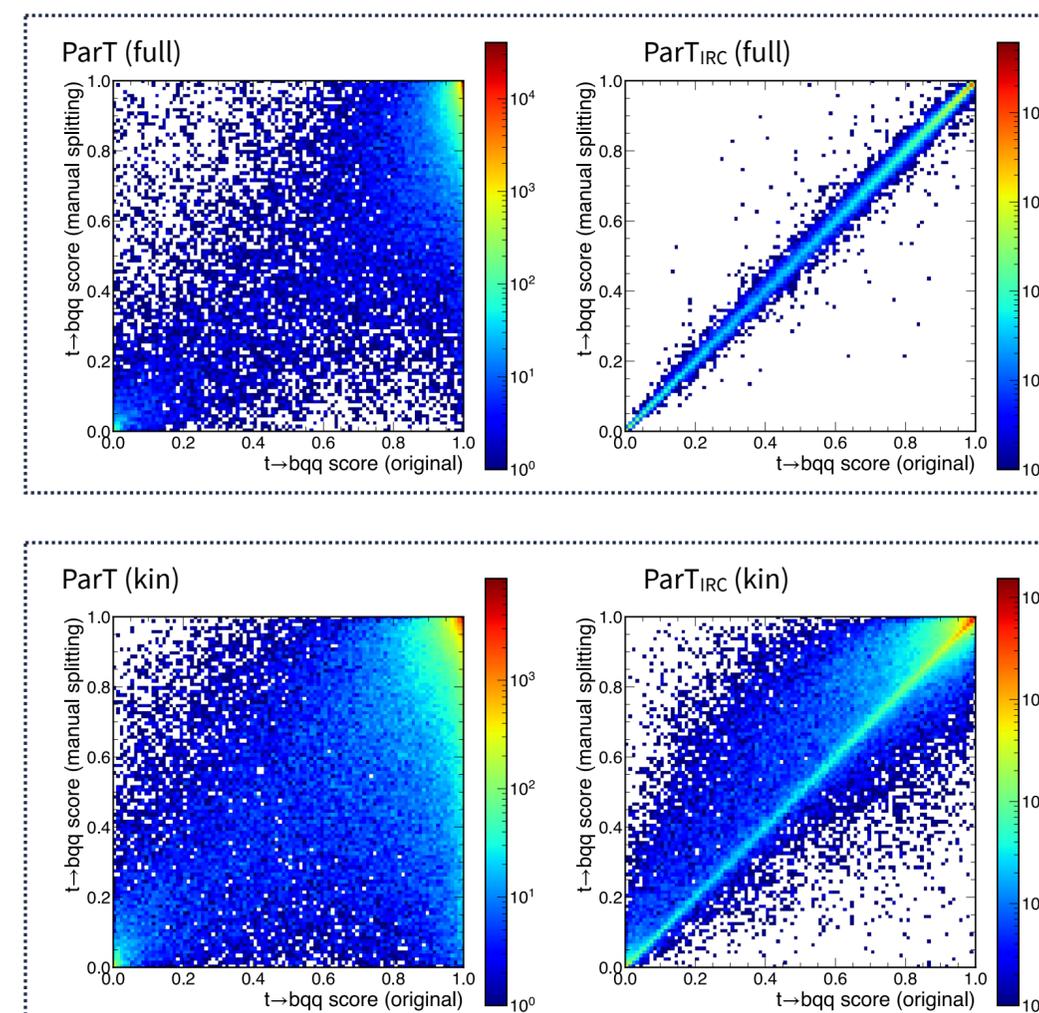
- **Overcoming systematic uncertainties:** can we build more robustness into the tagger design?
- Recent developments to **incorporate IRC safety into SOTA taggers** (PELICAN, ParT, ...)
 - at the cost of only a small loss in background rejection power
 - **but is IRC safety enough?**

See talks by [T. Hoffman](#) and [C. Li](#)

Performance on top tagging benchmark

Architecture	Accuracy	AUC	$1/\epsilon_B$	# Params
TopoDNN[48]	0.916	0.972	382 ± 5	59k
EFN[24]	0.927	0.979	729 ± 13	82k
LGN[25]	0.929(1)	0.964(14)	424 ± 82	4.5k
BIP(XGBoost)[49]	0.929	0.978	600 ± 47	312
EFP[18]	0.932	0.980	384	1k
BIP(MLP)[49]	0.931	0.981	853 ± 68	4k
PFN[24]	0.932	0.982	891 ± 18	82k
ResNeXt[8]	0.936	0.984	1122 ± 47	1.46M
ParticleNet[50]	0.938	0.985	1298 ± 46	498k
ParT[35]	0.940	0.9858	1602 ± 81	2.1M
LorentzNet[26]	0.942	0.9868	2195 ± 173	220k
PELICAN	0.9426(2)	0.9870(1)	2250 ± 75	208k
PELICAN _{IRC}	0.9406(2)	0.9844(11)	1711 ± 208	208k

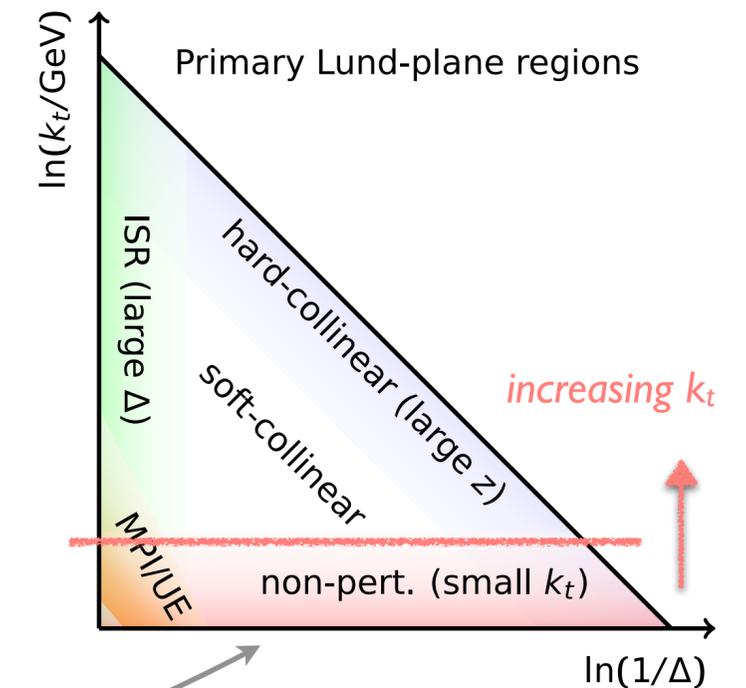
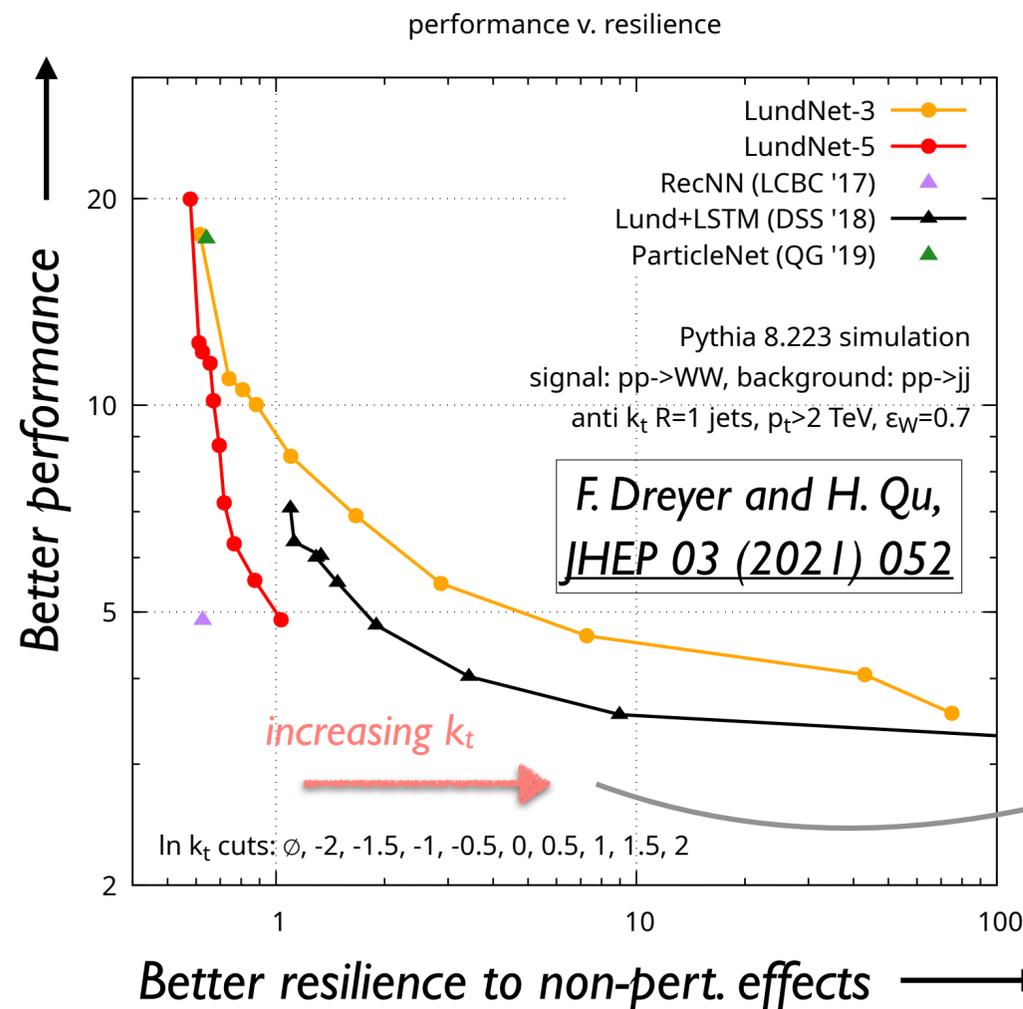
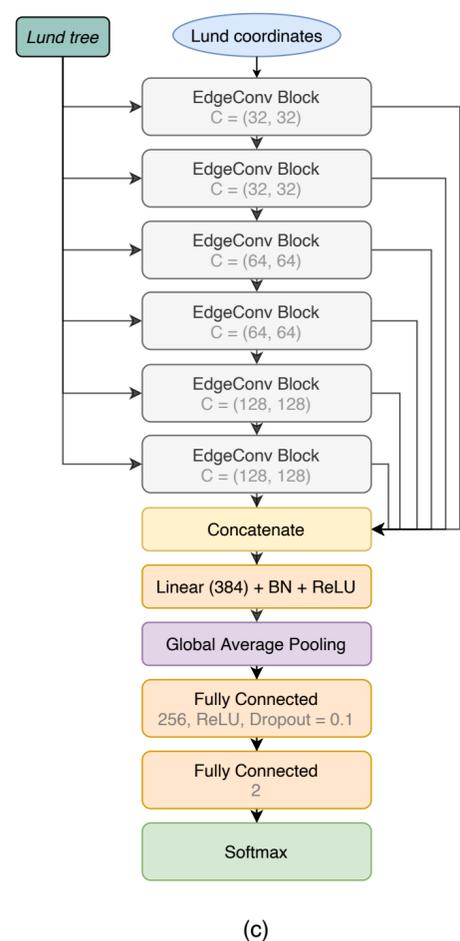
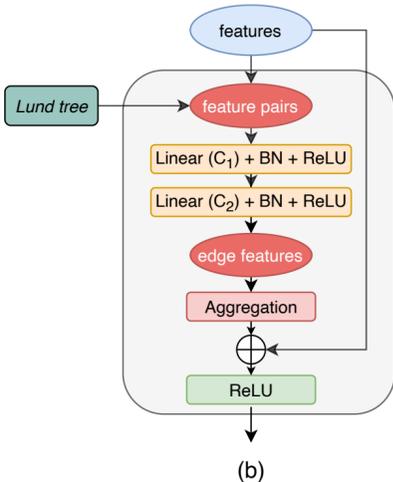
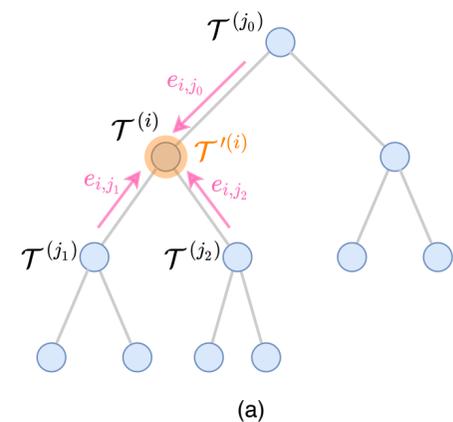
Testing IRC safety of ParT_{IRC}



A. Bogatskiy, T. Hoffman, D.W. Miller, J.T. Offermann and X. Liu, [2307.16506](#)

DESIGNING ROBUST TAGGERS (II)

- **Overcoming systematic uncertainties:** can we build more robustness into the tagger design?
- Recent developments to incorporate IRC safety into SOTA taggers (PELICAN, ParT, ...)
- Another approach is to use **physics-inspired representations** (e.g., the Lund jet plane)
 - still, is this sufficient?
 - how to incorporate flavor information (e.g., track displacement) into the Lund plane / tree representation?



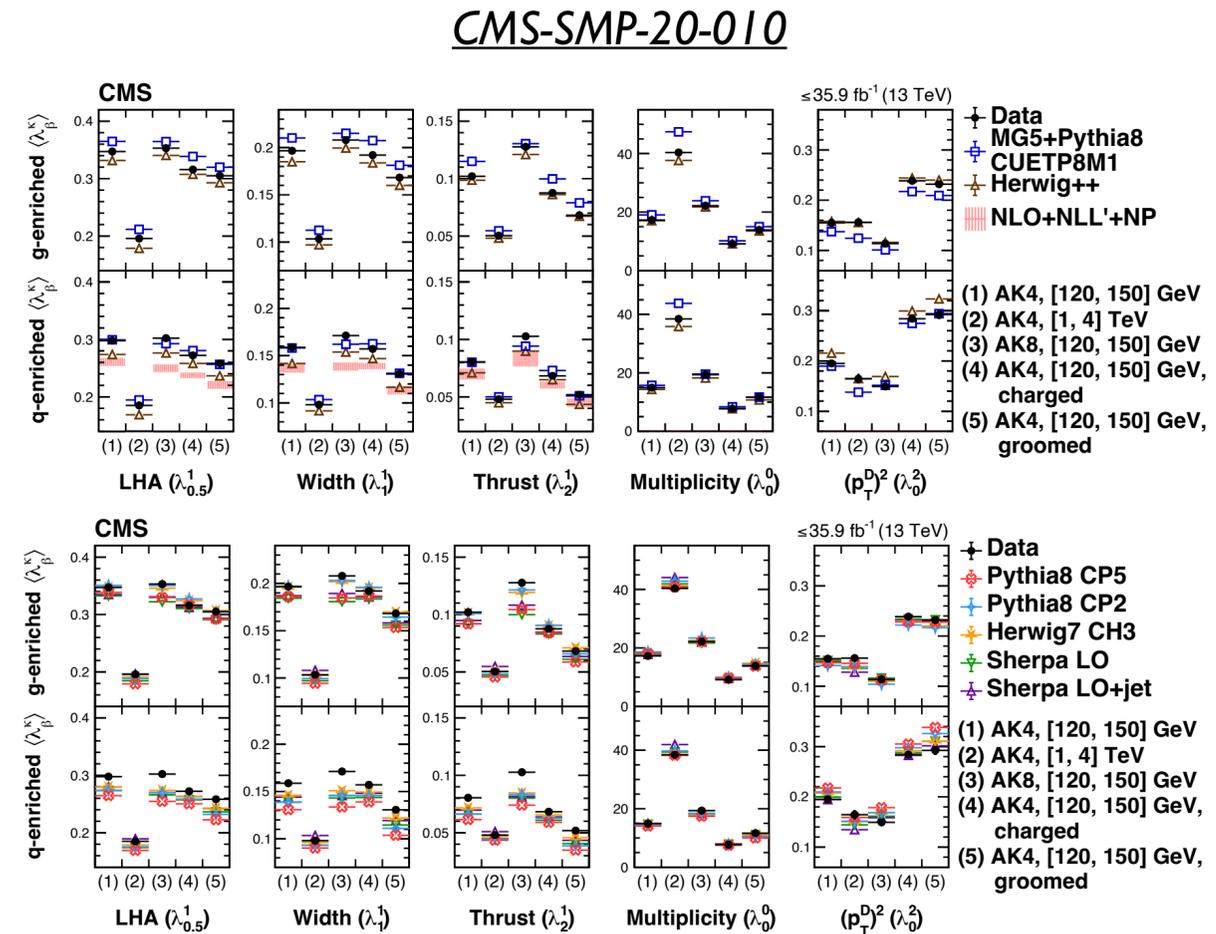
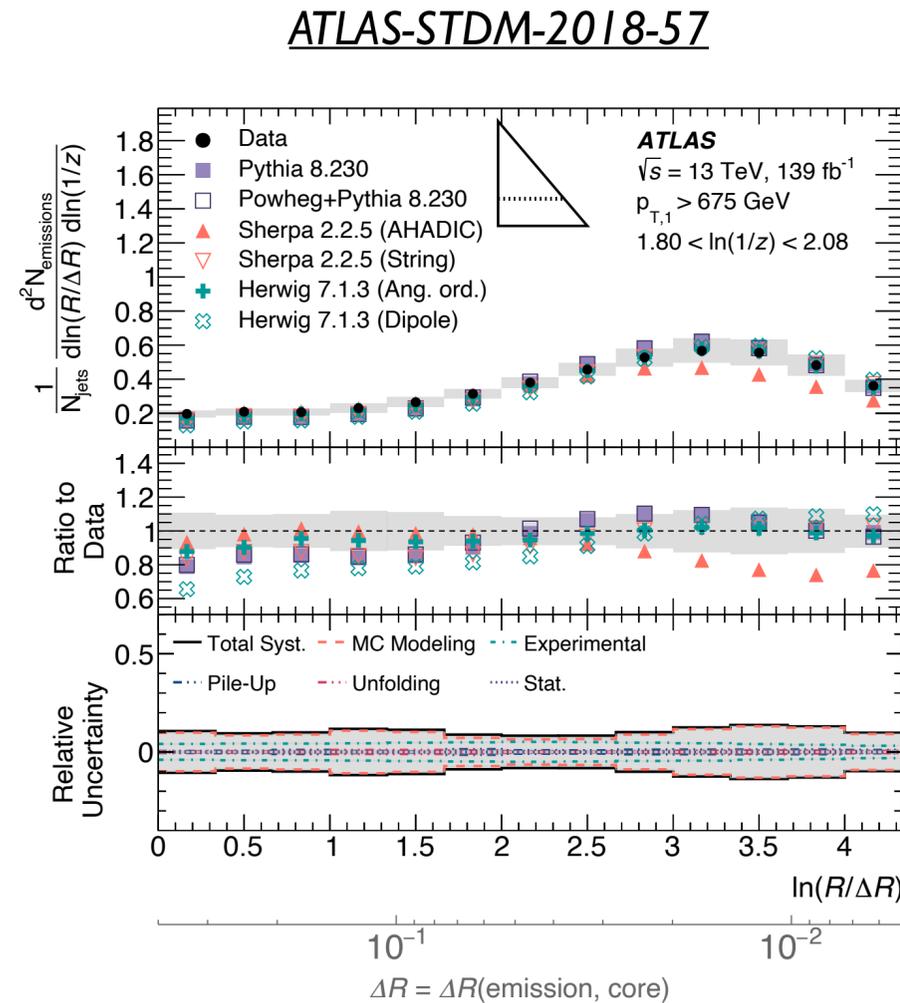
* Resilience assessed by applying the model trained on hadron-level samples to parton-level samples and compare the difference

DESIGNING ROBUST TAGGERS (III)

- **Overcoming systematic uncertainties:** can we build more robustness into the tagger design?
- Recent developments to incorporate IRC safety into SOTA taggers (PELICAN, ParT, ...)
- Another approach is to use physics-inspired representations (e.g., the Lund jet plane)
- Or, can we achieve better robustness via **alternative training techniques**?
 - adversarial attacks (e.g., 2203.13890)? data augmentation?
 - **but what are the physically motivated perturbations / augmentations?**

UNCERTAINTY & ROBUSTNESS

- The ultimate solution: understand the discrepancies and improve the shower/hadronization modeling



- Joint effort between theory, experiment, and ML communities
 - some discussions started in the Les Houches workshop this year – get in touch if you are interested

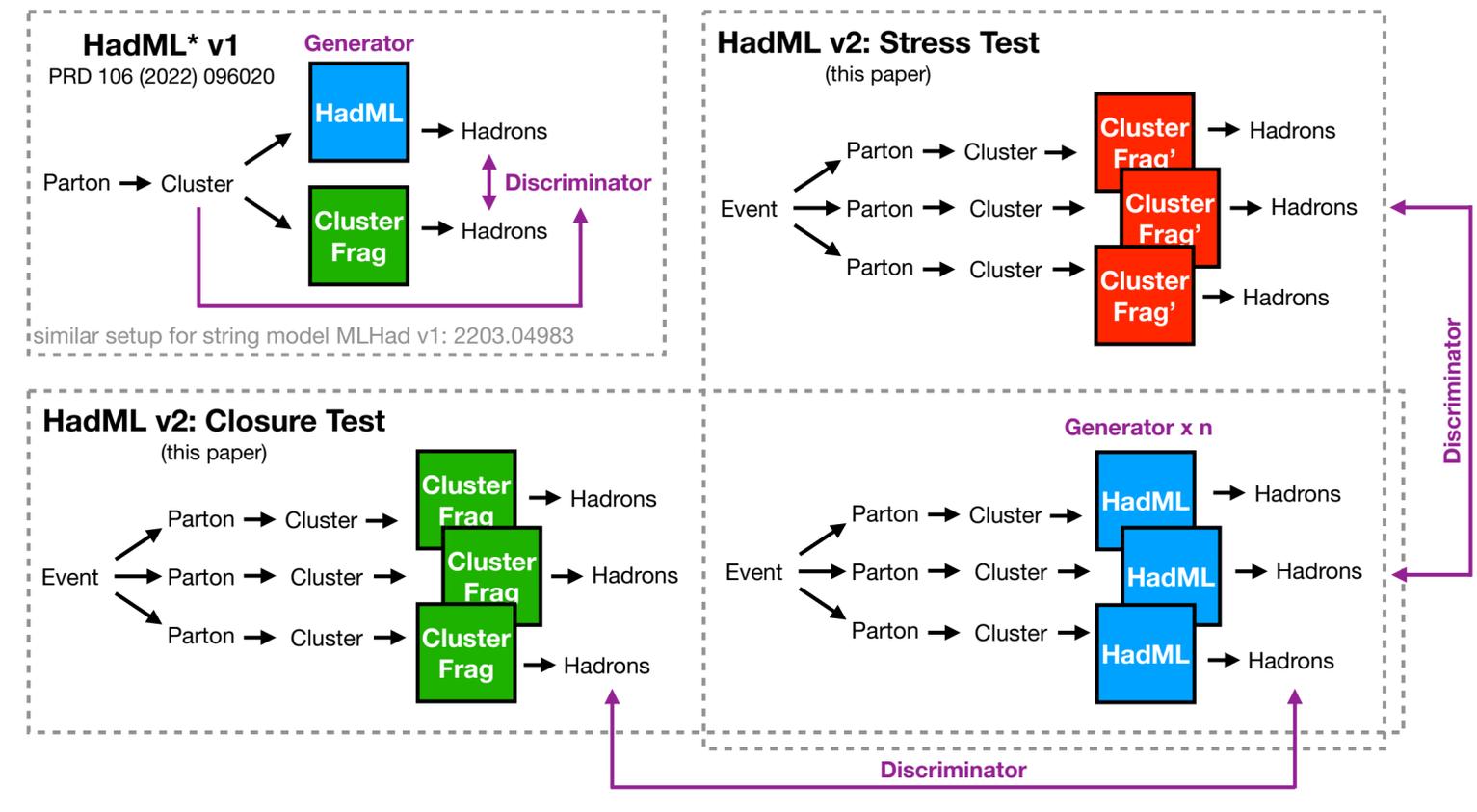
UNCERTAINTY & ROBUSTNESS

- The ultimate solution: **understand the discrepancies and improve the shower/hadronization modeling**
- Can we use ML to solve problems induced by the usage of ML?
 - probably yes!

JHEP 09 (2023) 084

Fitting a deep generative hadronization model

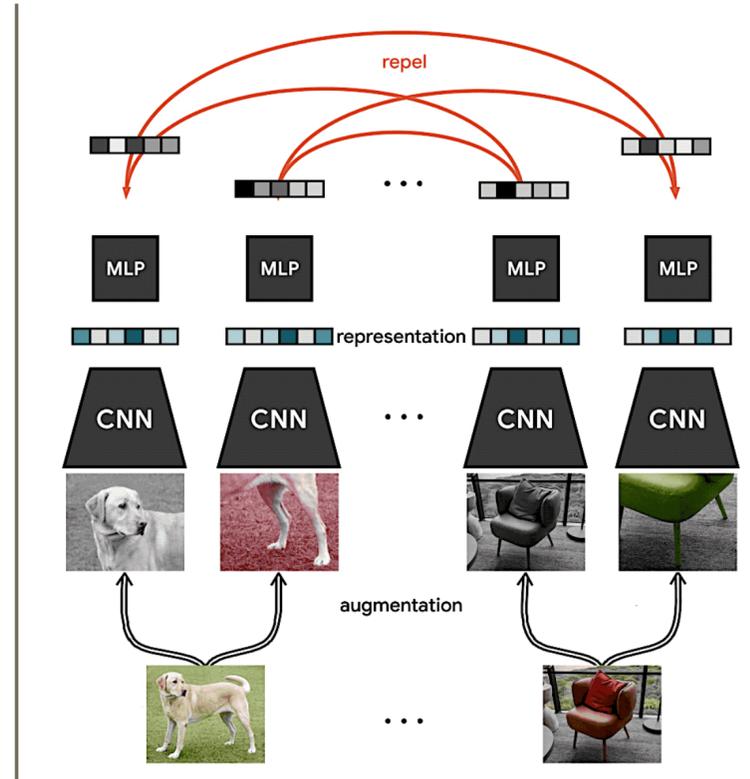
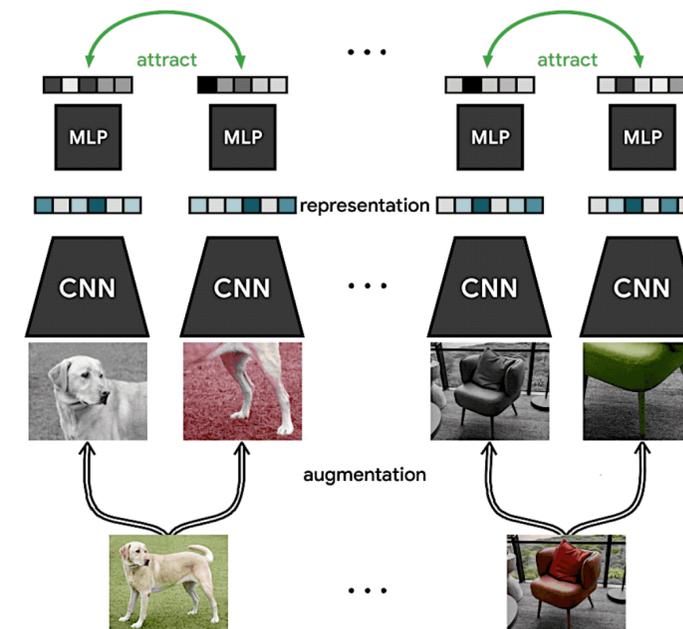
Jay Chan,^{a,b} Xiangyang Ju,^c Adam Kania,^f Benjamin Nachman,^{b,d} Vishnu Sangli^{e,b}
and Andrzej Siódmok^f



See talks by Adam Kania

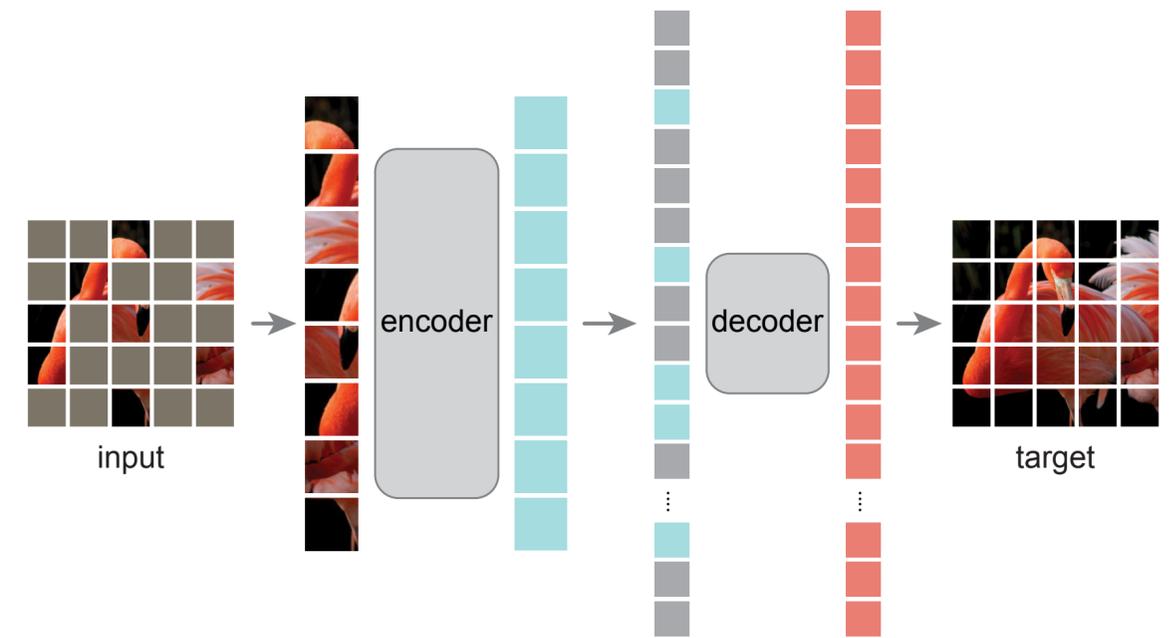
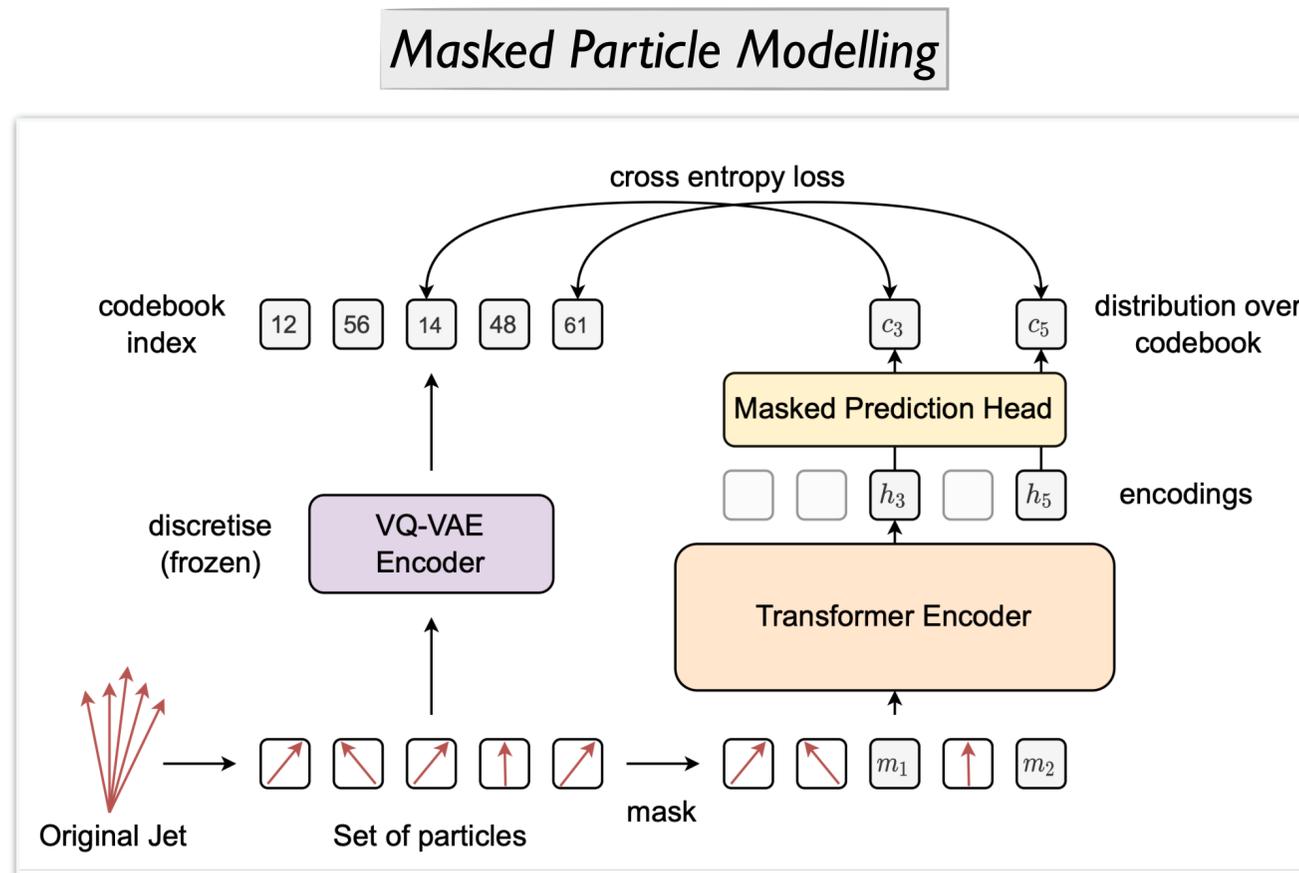
SELF-SUPERVISED LEARNING

- Self-Supervised Learning (SSL): Can we rely less on (labelled) simulation to achieve more?
 - learn generic representations for all kinds of downstream tasks: tagging, reconstruction, anomaly detection, ...
- **Contrastive learning (SimCLR)**
 - maximize similarity between positive pairs and minimize that between negative pairs
 - positive pairs derived from the same sample, but with different augmentations (cropping, color distortion, ...)
- **But what is the most effective augmentation for jets?**
 - JetCLR [2108.04253]: rotation, translation, soft/collinear splitting
 - re-simulation as an augmentation
 - re-simulate w/ varied shower/hadronization models/configs
 - talk by [J. Krupa](#) @ BOOST 2023
 - re-simulate w/ varied detector configurations
 - talk by [E. Dreyer](#) on Monday
 - potentially learning a more robust representation
 - but not applicable to real data...



SELF-SUPERVISED LEARNING (II)

- Self-Supervised Learning: Can we rely less on simulation/labels to achieve more?
- Contrastive learning (SimCLR)
- **Masked models (BERT/MAE)**
 - learning to recover randomly masked tokens (words/pixels => particles)
 - straightforwardly applicable to real data



See talk by [M. Leigh](#) on Monday

SELF-SUPERVISED LEARNING (III)

- Self-Supervised Learning: Can we rely less on simulation/labels to achieve more?
- Contrastive learning (SimCLR)
- Masked models (BERT/MAE)
- **Autoregressive (GPT)**
 - predict the next word based on previous words
 - conceptually not so natural as particles in jets are unordered
 - but what if we impose an ordering?

JHEP 06 (2023) 184

Learning the language of QCD jets with transformers

Thorben Finke,^a Michael Krämer,^a Alexander Mück^a and Jan Tönshoff^b

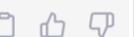
See talk by A. Mück on Tuesday



How does GPT work? Please answer in one paragraph.



GPT (Generative Pre-trained Transformer) is a type of artificial neural network that utilizes a transformer architecture to understand and generate human-like text. It works by pre-training on a massive corpus of text data, learning to predict the next word in a sentence and gaining an understanding of language structure and semantics. During pre-training, it learns to create contextual word embeddings, which capture the relationships between words in a sentence. These embeddings are then fine-tuned on specific tasks, such as text generation or language translation. GPT generates text by sampling words or phrases based on the context it has learned, making it capable of completing sentences, answering questions, and generating coherent and contextually relevant text.



SELF-SUPERVISED LEARNING (IV)

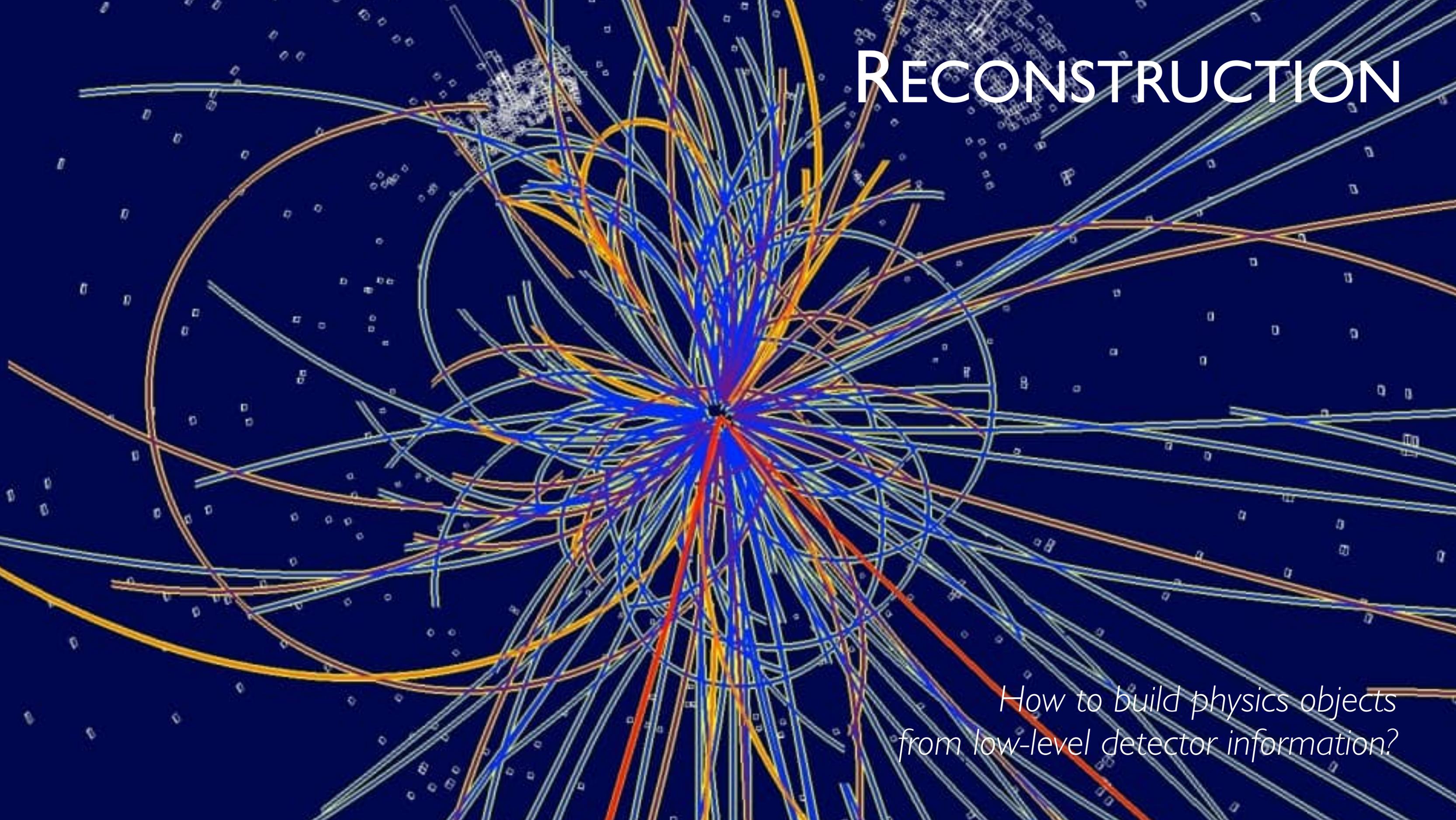
- Self-Supervised Learning: Can we rely less on simulation/labels to achieve more?
- Contrastive learning (SimCLR)
- Masked models (BERT/MAE)
- Autoregressive (GPT)

- **Some open questions for masked/autoregressive models**
 - how to best embed continuous particle features for masked/autoregressive models?
 - unlike images, particle features (such as momentum/energy) can span several orders of magnitude
 - maybe the CDFDequantization proposed in CaloPointFlow II (see talk by [S. Schnake](#))?
 - how to deal with permutation invariance?
 - what is the best ordering for autoregressive model?
 - how to break the degeneracy when predicting more than one masked particles?

SELF-SUPERVISED LEARNING (V)

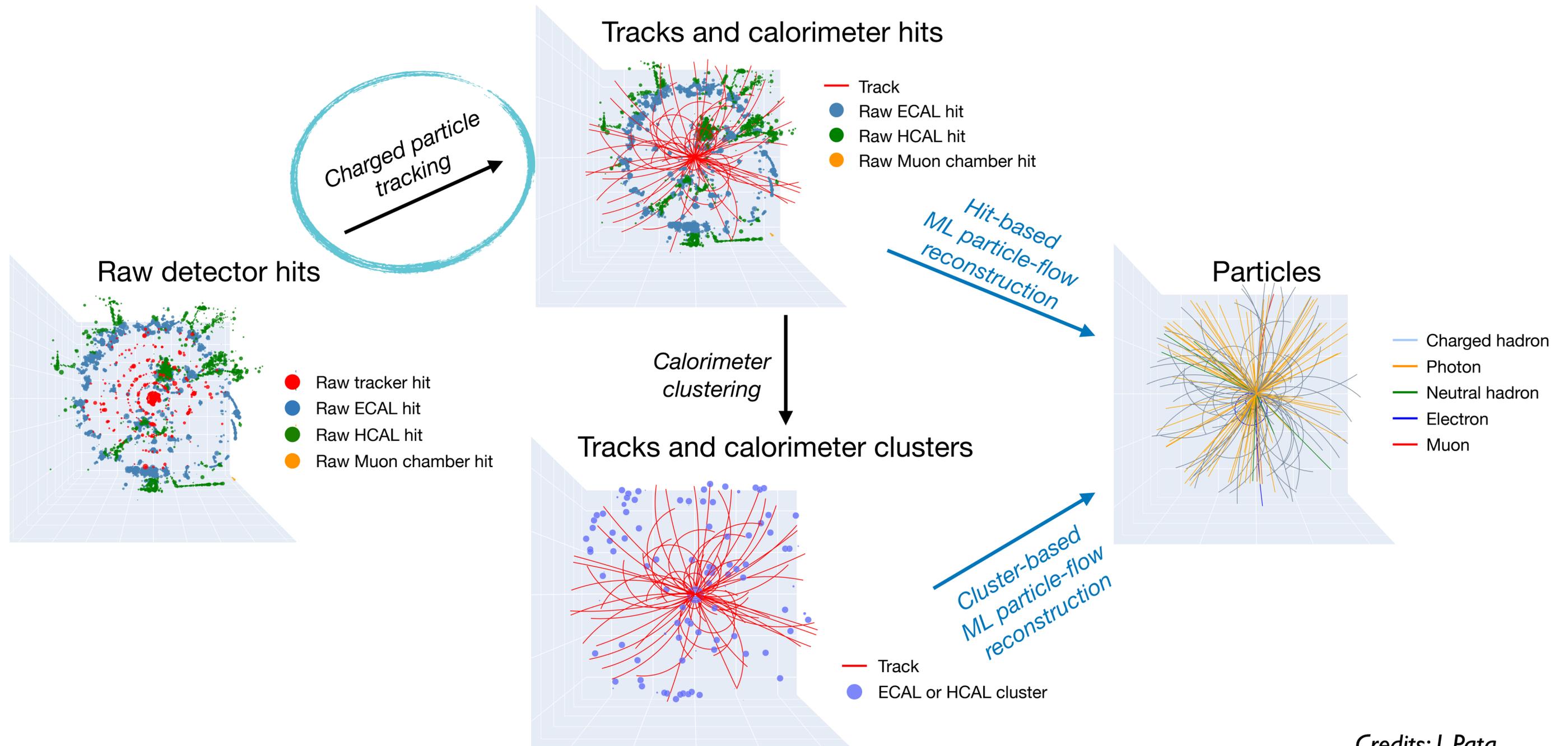
- Self-Supervised Learning: Can we rely less on simulation/labels to achieve more?
- Contrastive learning (SimCLR)
- Masked models (BERT/MAE)
- Autoregressive (GPT)
- ...
- More ideas?
- Rising interests in self-supervised learning and foundation models
 - the next ML4Jets challenge?

RECONSTRUCTION



*How to build physics objects
from low-level detector information?*

RECONSTRUCTION

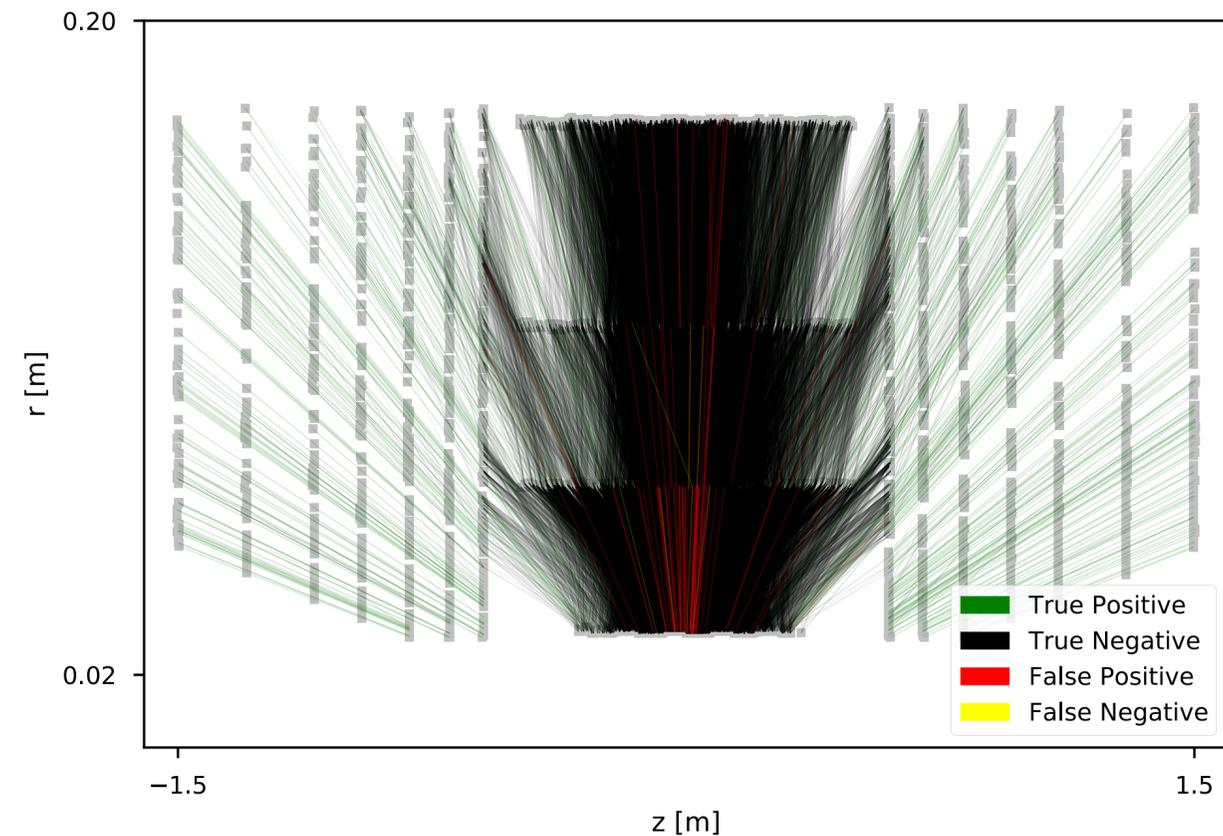
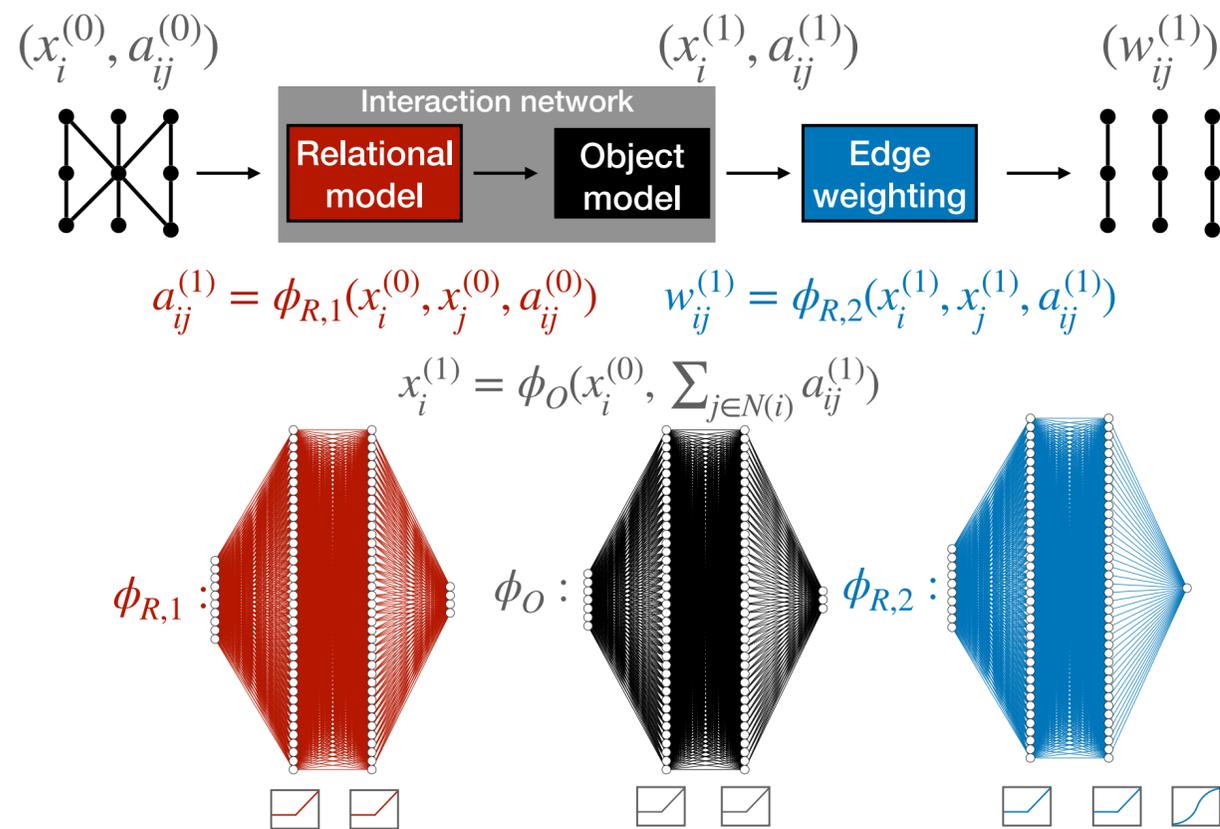


Credits: J. Pata

CHARGED PARTICLE TRACKING

- Charged particle tracking via **edge classification** with GNNs
 - each hit is a node of the graph
 - edges constructed between pairs of hits with geometrically plausible relations
 - classify whether each edge connects hits belonging to the same track or not

G. DeZoort et al.
 [Comput. Softw. Big Sci. 5, 26 (2021)]



See also: S. Farrell et al. [1810.06111]; X. Ju et al. [2003.11603];
 C. Biscarat, S. Caillou, C. Rougier, J. Stark and J. Zahreddine [2103.00916]; X. Ju et al. [2103.06995]; etc.

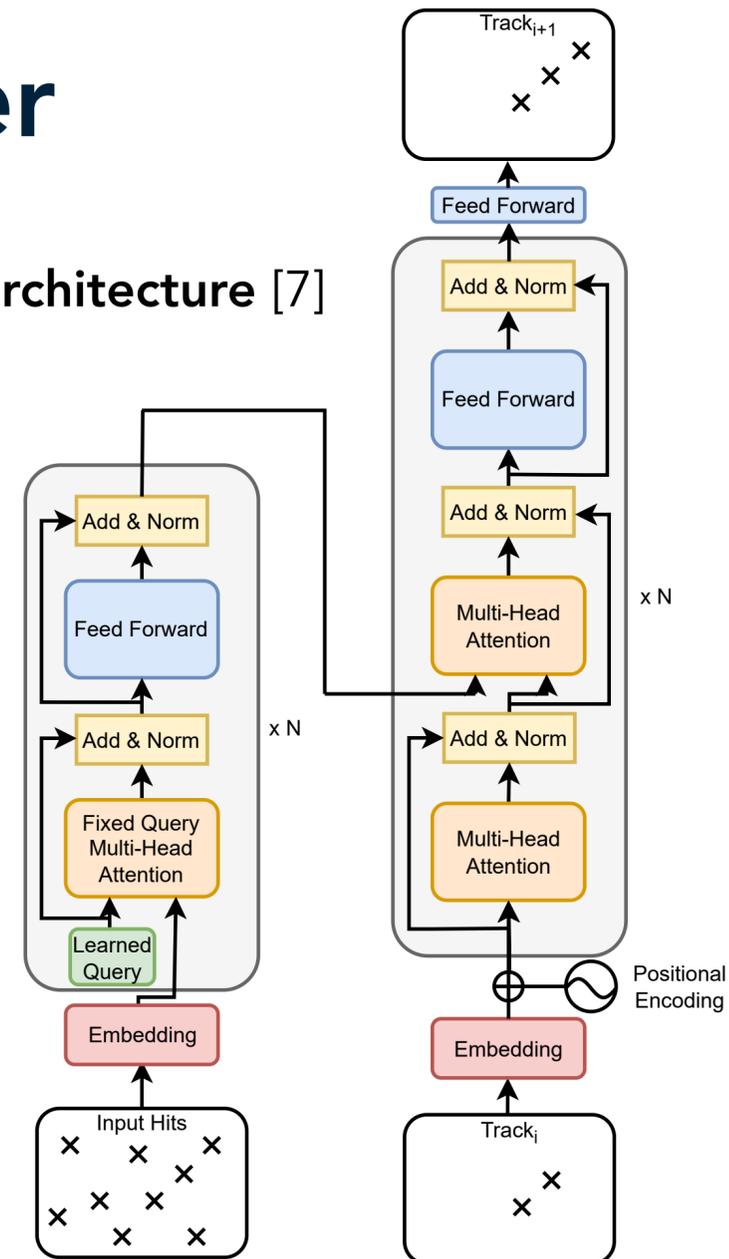
CHARGED PARTICLE TRACKING (II)

- New approach: encoder-decoder architecture based on Transformer

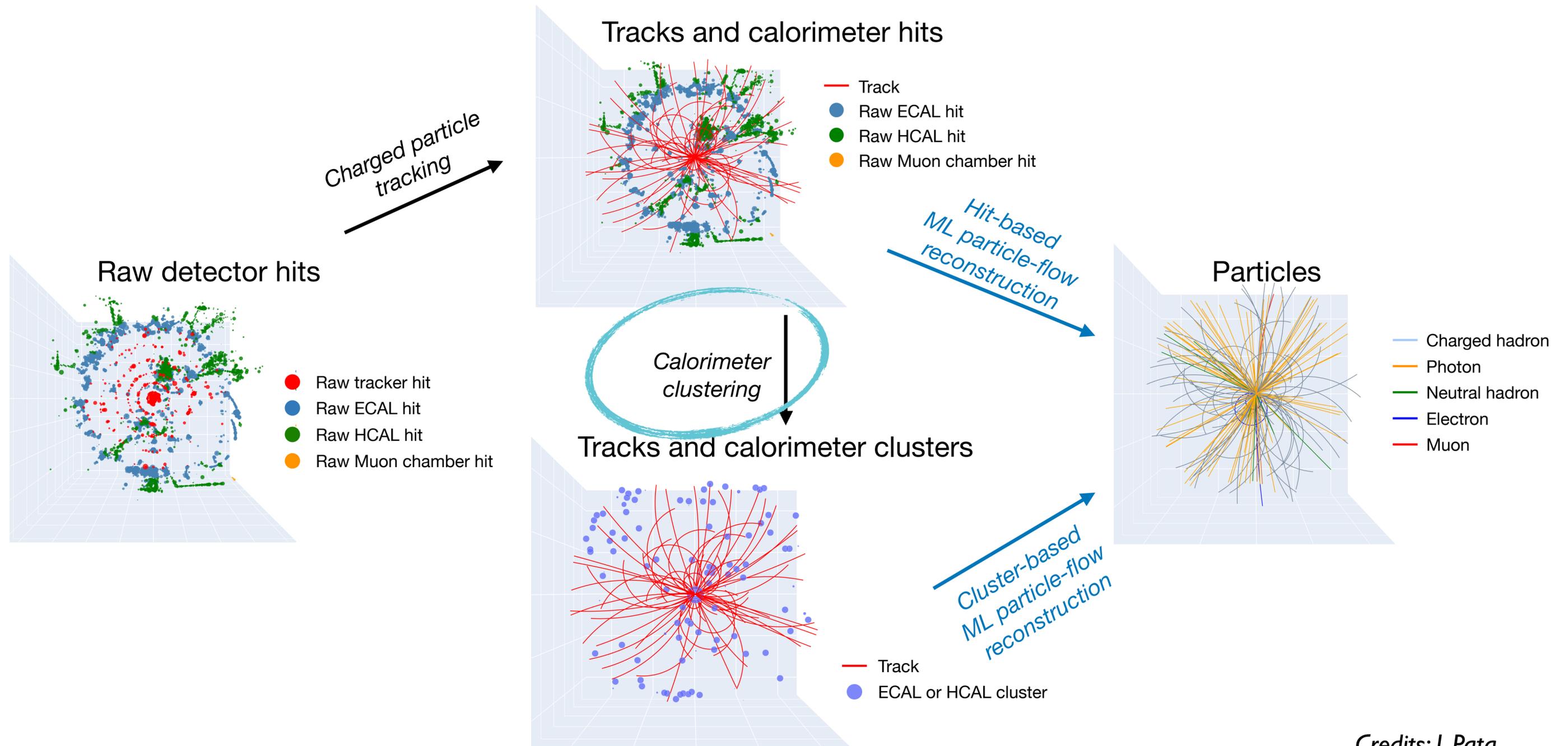
Talk by Zef Wolffs

Transformers - Trackformer

- This model resembles closely the **original transformer architecture** [7]
- Translating, e.g. English to Spanish, is a typical task for transformer models
 - This model in similar fashion **translates hits to tracks**
- **Encoder:** Encodes full event hits
 - **No positional embedding** as hits have no particular order
 - **Fixed-query attention** [8] to achieve full positional invariance of inputs
- **Decoder:** Predicts next hit in track
 - **Autoregressively builds the full track**, starting from a given seed



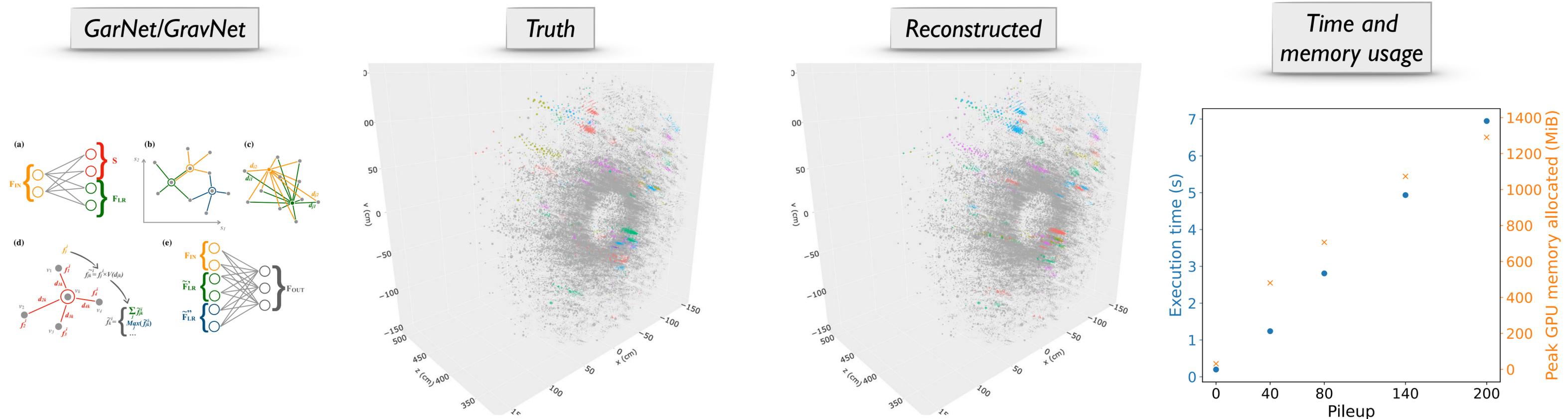
RECONSTRUCTION



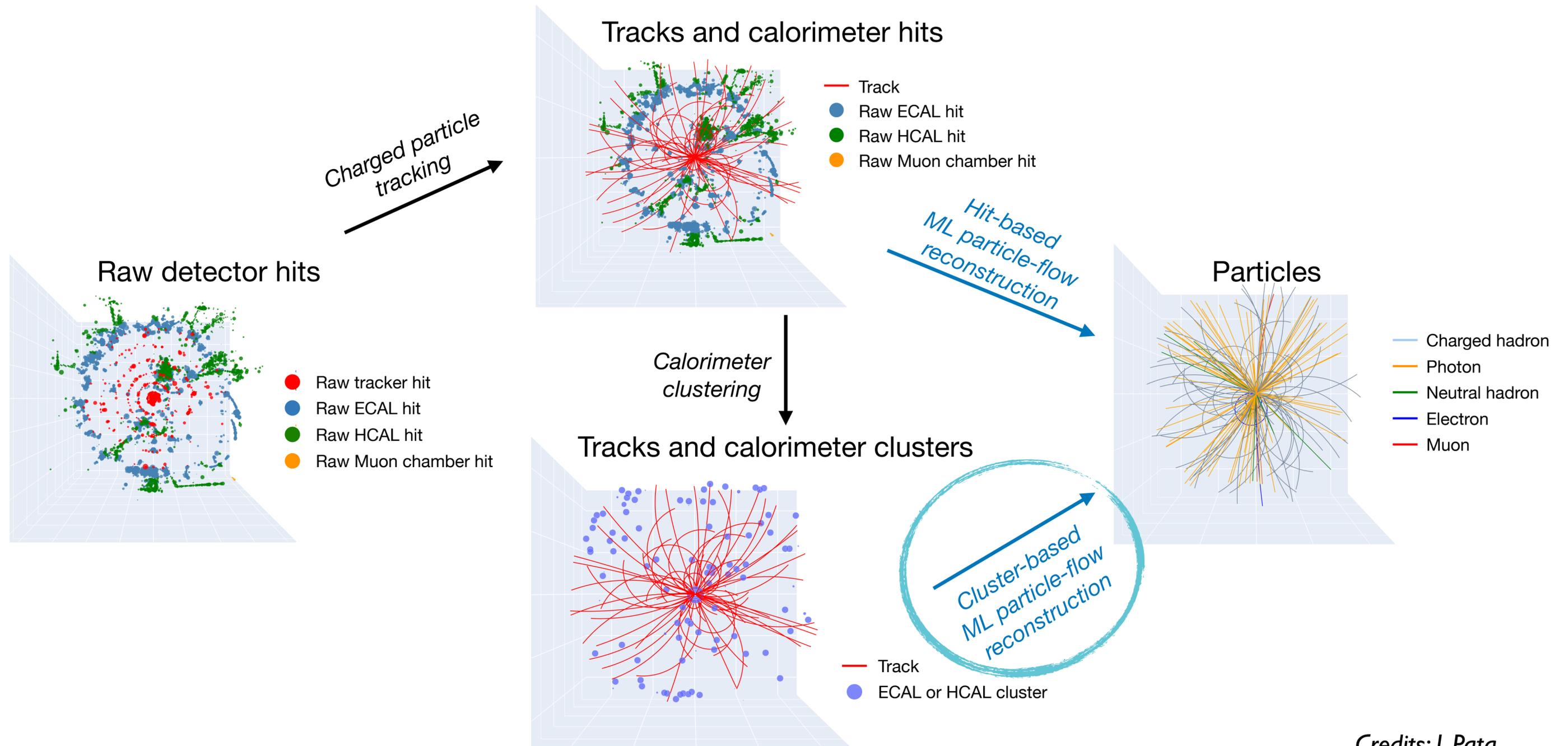
Credits: J. Pata

CALORIMETER CLUSTERING

- End-to-end reconstruction for a high granularity calorimeter
- Object condensation: one-stage multi-object reconstruction
 - supervised clustering of hits belonging to a shower to a “condensation point” by using attractive/repulsive potentials in the loss
 - simultaneously predict the number of showers and their properties



RECONSTRUCTION

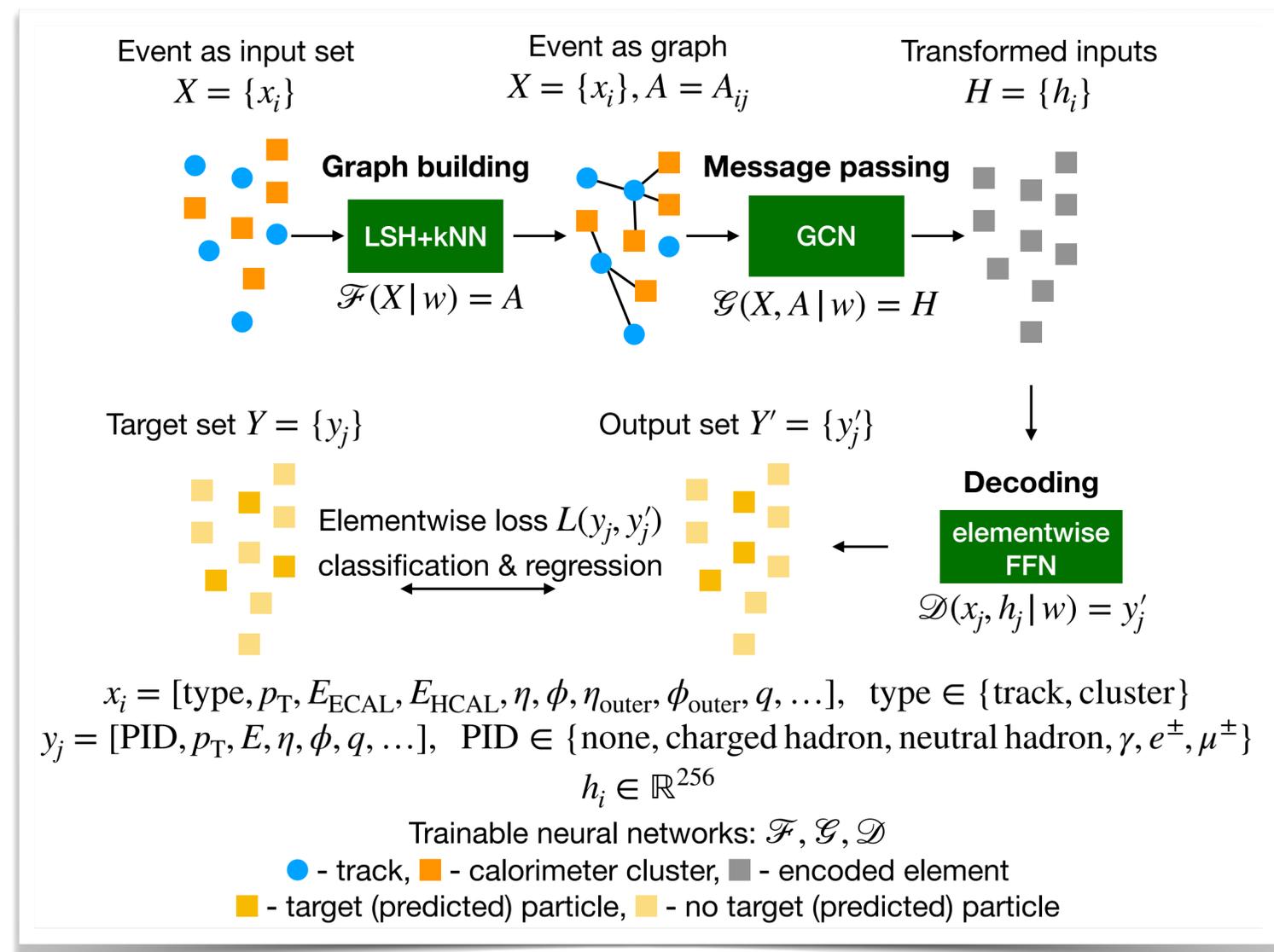
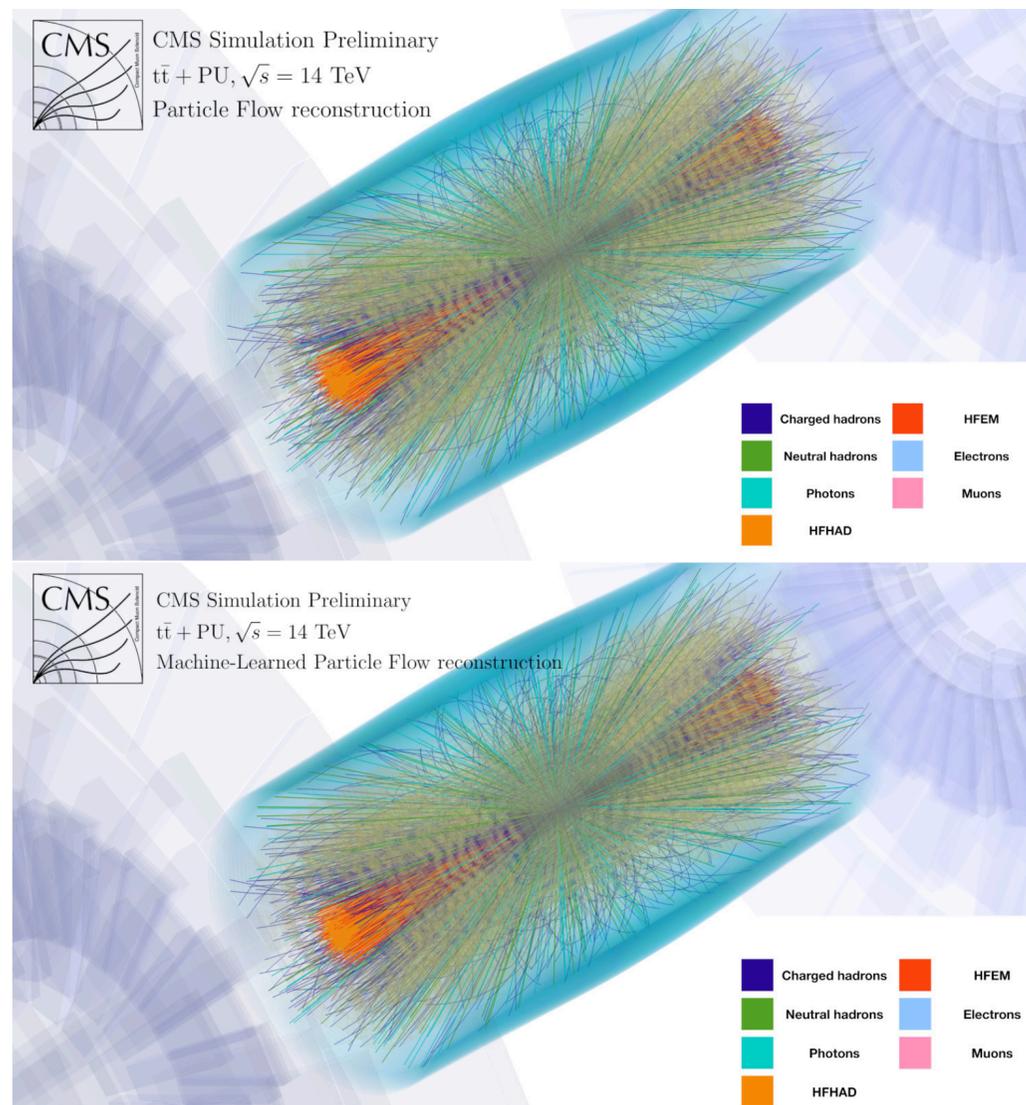


Credits: J. Pata

PARTICLE-FLOW RECONSTRUCTION: MLPF

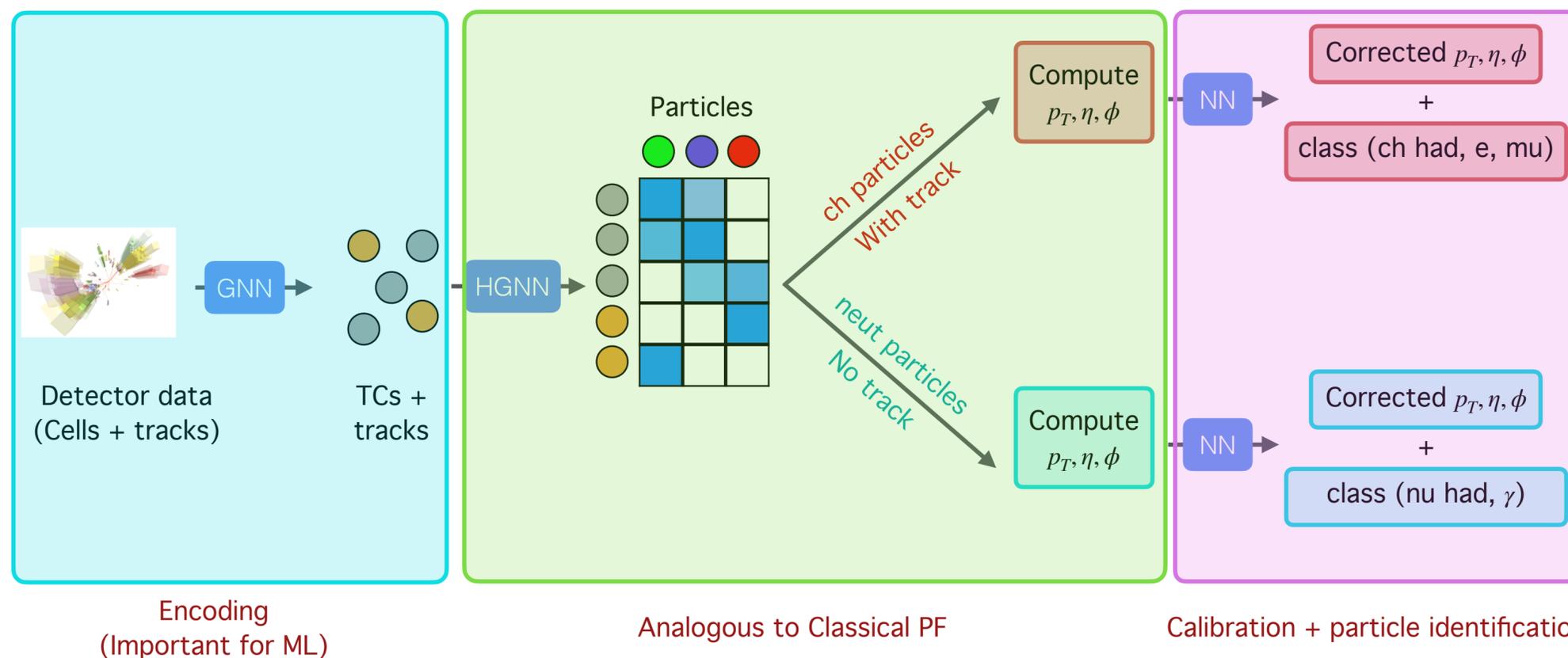
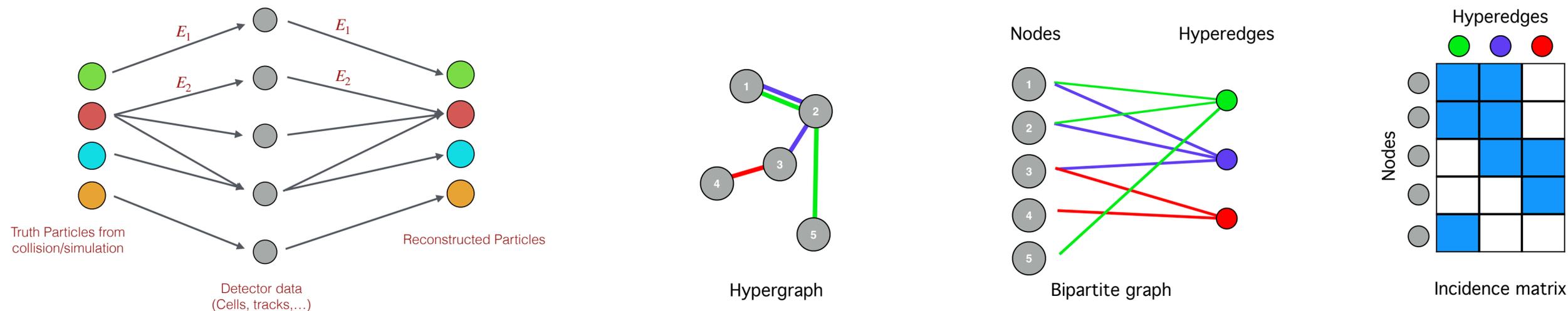
- Global particle-flow reconstruction via node classification and regression using GNNs
 - comparable performance to the baseline rule-based PF algorithm at CMS
 - runtime scales linearly with input size, no quartic explosion

*J. Pata et al.,
ACAT 2021, 2203.00330*



PARTICLE-FLOW RECONSTRUCTION: HGPFLOW

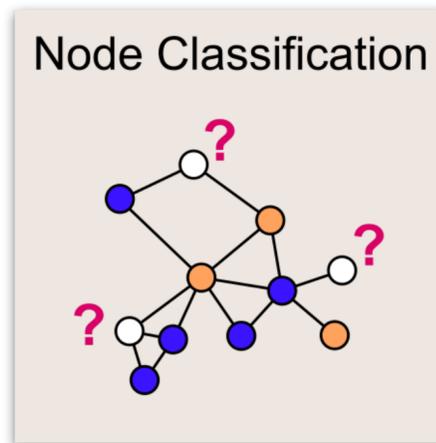
- What if multiple clusters should be associated to multiple particles? ==> Hypergraph



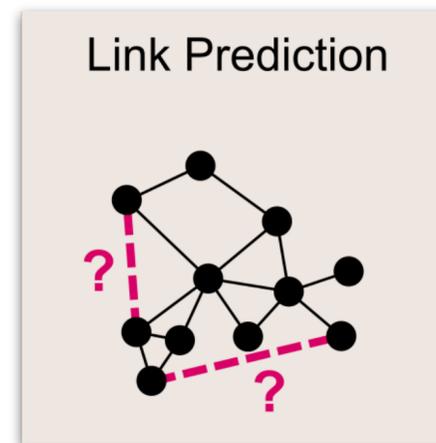
See talk by N. Kakati on Monday

ML FOR RECONSTRUCTION

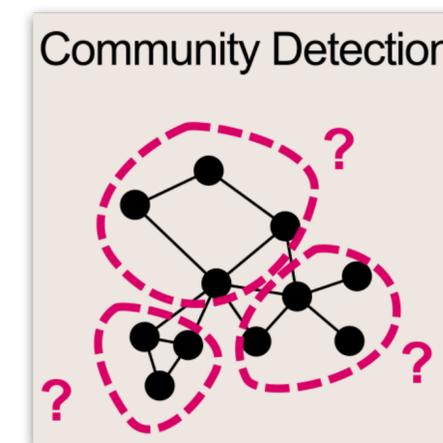
- Almost all reconstruction models are based on point clouds and using graph networks / Transformers
 - however, with varying approaches for different tasks



MLPF



*Tracking,
HGPflow*

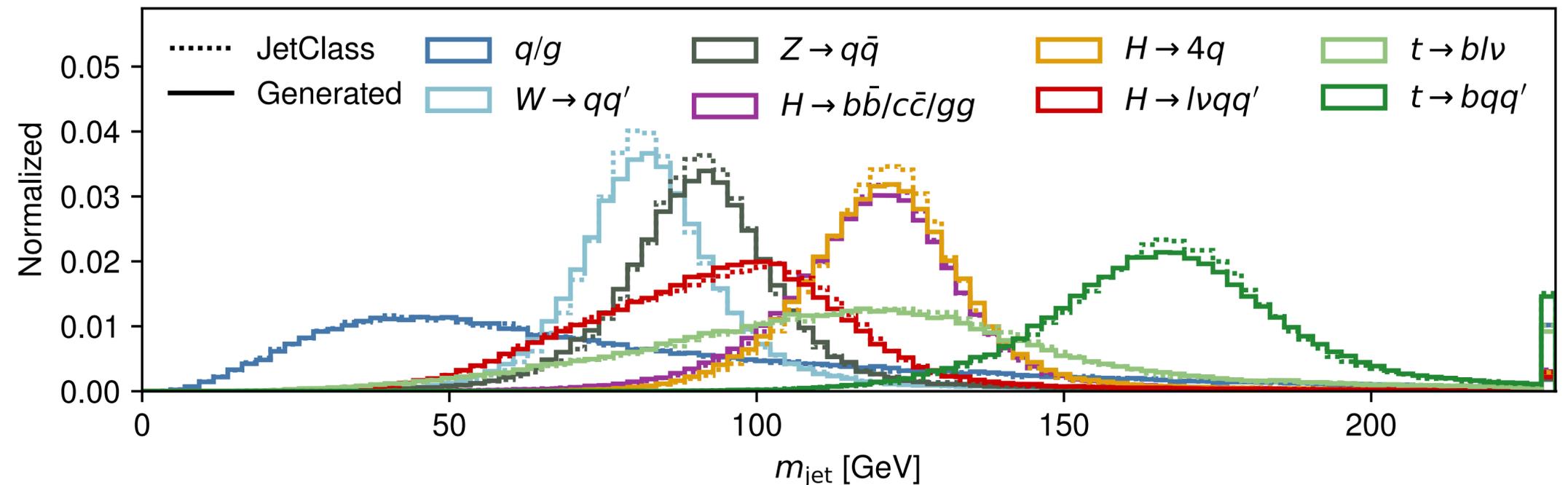
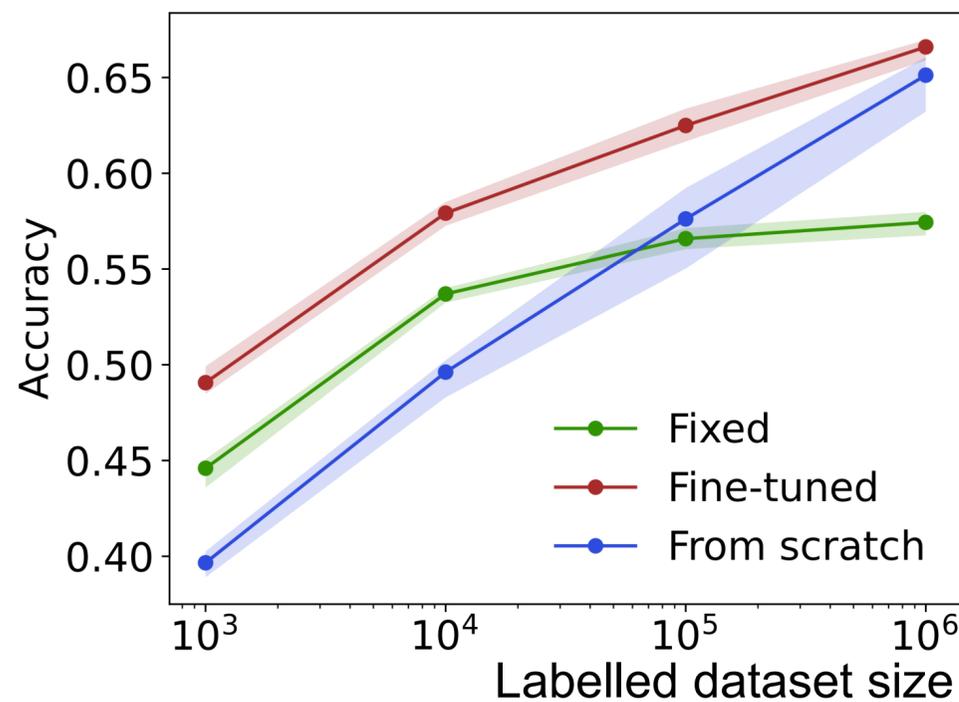


*Calo clustering
(object condensation)*

- Open questions:
 - can we unify the approach for all task? Or each task is different?
 - so far nearly all models are fully supervised
 - requiring highly detailed simulation truth information, which are not always so straightforward to define
 - any possibilities for less supervision? Self-supervision? Optimal Transport? ...?
 - leading toward future colliders – interplay with detector optimization? Another area for software-hardware co-design?

OPEN DATASETS AND BENCHMARKS

- Open datasets and common benchmarks are crucial for ML researches
 - and have been an extremely successful tradition of ML4Jets
 - top tagging landscape, LHCO anomaly detection, CaloChallenge, ...
 - nice to see large-scale datasets (such as [JETCLASS](#)) start to gain some traction in the community
 - self-supervised learning via “Masked Particle Modelling”: see talk by [Matthew Leigh](#)
 - generative models for jets beyond pure kinematics: see talk by [Joschka Birk](#)

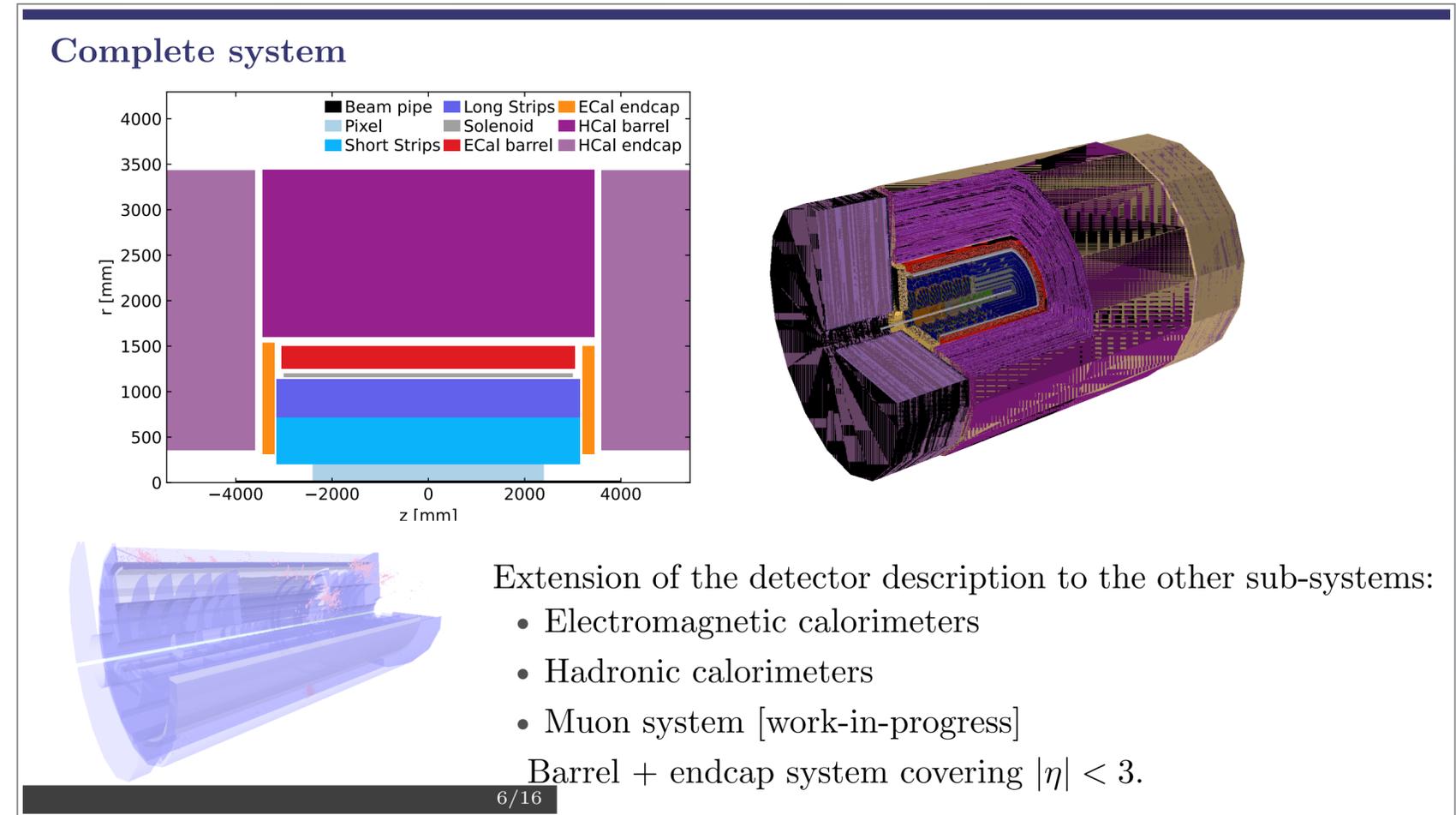
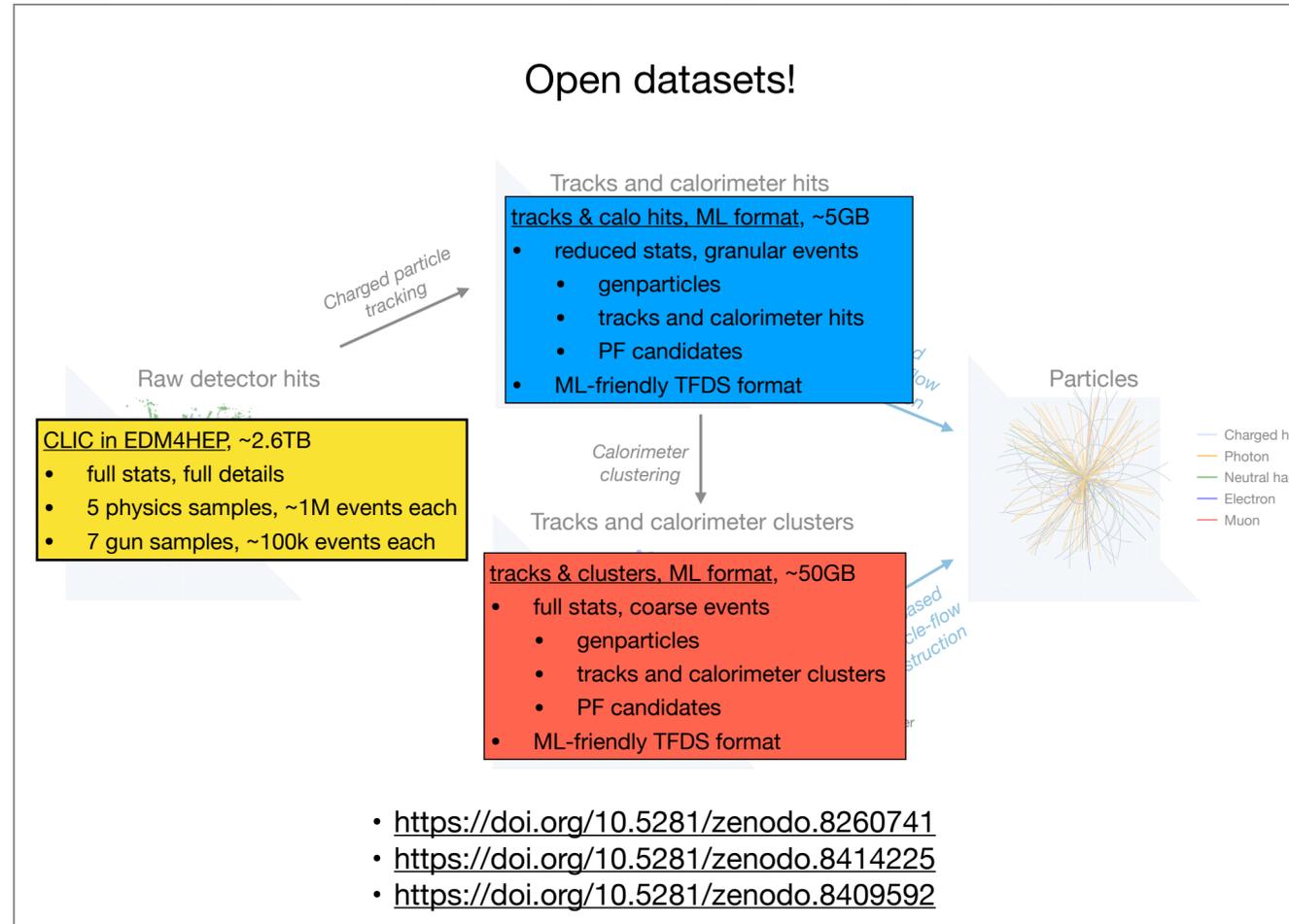


OPEN DATASETS AND BENCHMARKS (II)

- Even more open datasets coming this year
 - and not only datasets, but also open simulation / data generation toolchains ([COCOA](#), [REDVID](#), [ODD](#), ...)

See talk by [J. Pata](#)

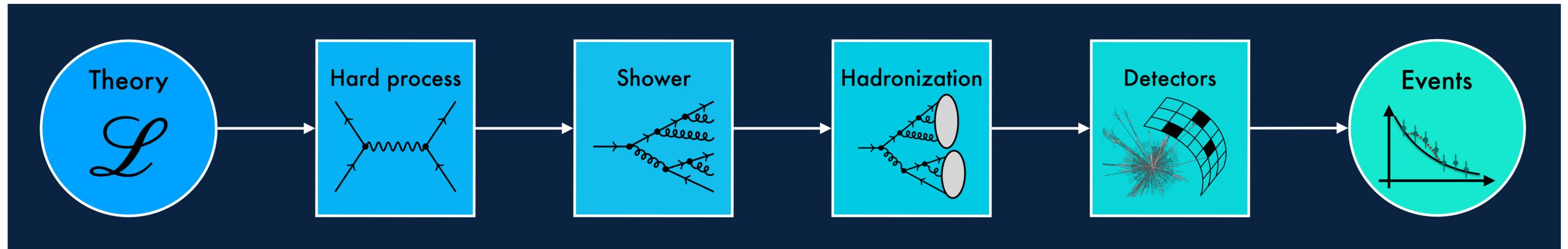
See talk by [A. Zaborowska](#)



OUTLOOK: A *UNIFIED FLOW*?

Generation, Simulation, ...

Forward



Inverse

Reconstruction, Unfolding, ...

Credits: R. Winterhalder

- A foundation model for both directions?
 - or less ambitiously:
 - a common representation / (pre-trained) backbone for subsets of the problems, such as simulation vs. reconstruction?